

NPS ARCHIVE
1966
HUDSON, R.

FILTERING NONLINEAR MEASUREMENTS

RALPH E. HUDSON

LIBRARY
NAVAL POSTGRADUATE SCHOOL
MONTEREY, CALIF. 93943-5101

DUDLEY KNOX LIBRARY
NAVAL POSTGRADUATE SCHOOL
MONTEREY CA 93943-5101

DUDLEY KNOX LIBRARY
NAVAL POSTGRADUATE SCHOOL
MONTEREY CA 93943-5101

FILTERING NONLINEAR MEASUREMENTS

by

Ralph E. Hudson
Lieutenant, United States Navy
B.S., University of California, 1960

Submitted in partial fulfillment
for the degree of

DOCTOR OF PHILOSOPHY

from the

UNITED STATES NAVAL POSTGRADUATE SCHOOL
December 1966

1966
Hudson, R.

~~THESIS~~
~~H 8304~~
~~C.1~~

ABSTRACT

An algorithm is developed for estimating the state of a linear dynamic system excited by a random sequence. The input data are noisy observations which are nonlinear functions of the state. The estimates are best in the sense of least squared residuals. A significant problem in radar tracking is investigated and the effectiveness of the algorithm verified.

TABLE OF CONTENTS

Section	Page
1. Introduction	9
2. Detailed Statement of the Problem	12
3. Prior Work	20
4. Development of the Solution Algorithm	26
5. The Single-Stage Minimization Procedure	27
6. Multi-stage Case Cast in Single-stage Form	32
7. Criteria for Termination of the Minimization Procedure	38
8. Criteria for the Number of Smoothing Stages to be Carried	42
9. Computation for the Solution	45
10. A Simple Example	52
11. Target Tracking	62
12. Conclusions	76
Appendix	
I. Discussion of the Singularity of P_0	78
II. First and Second Moments of the Second-order Terms	83
III. Program Listings	88

LIST OF TABLES

Table		Page
I.	SINGLE-STAGE MINIMIZATION PROCESS	55
II.	ITERATION CONTROL PARAMETERS AS A FUNCTION OF THE NUMBER OF STAGES CARRIED FOR $\alpha = 0.05$ AND $\beta = 0.95$	68
III.	TYPICAL SEQUENCE OF COSTS AS A FUNCTION OF NUMBER OF ITERATIONS AND NUMBER OF OBSERVATIONS PROCESSED	73
IV.	TYPICAL COMPUTATIONAL TIME REQUIREMENTS FOR THE CDC 1604 COMPUTER	74

LIST OF ILLUSTRATIONS

Figure	Page
1. Flow Graph of the Criteria for Iteration Termination	40
2. Flow Graph of Overall Procedure	46
3. Flow Graph of Step-by-step Minimization Procedure	49
4. Evolution of the Estimate and Covariance During the Transition	50
5. Projection on NORTH-EAST Plane of True Trajectory with Estimates	69
6. Projection on NORTH-DOWN Plane of True Trajectory with Estimates	69
7. Observation and Estimation Errors in NORTH Coordinate	70
8. Observation and Estimation Errors in EAST Coordinate	71
9. Observation and Estimation Errors in DOWN Coordinate	71

TABLE OF SYMBOLS

Scalar	Meaning	
α	Probability that a minimum cost has a value greater than C_L	
β	Significance level for comparing iterated estimates	
C	Sum of squares of residuals, referred to as cost	
k	Time sequence index	
m	Number of independent noise inputs to dynamic system	
n	Order of dynamic system	
q	Step size reduction factor	
r	Number of observations made at each stage Also rank of a matrix	
σ	Standard deviation	
Vector		Dimension
d	Computed step without overshoot control	(n)
D	Actual step after overshoot control	(n)
$f(.,.)$	Vector valued function of the state and noise	(n)
g	Gradient of cost with respect to estimate	(n)
$h(.)$	Vector valued observation function of the state	(r)
v	Noise added to observation	(r)
w	Noise disturbing dynamic system	(m)
x	State	(n)
z	Observation	(r)

Matrix

G	Filter adjustment matrix	(m x r)
Γ	Input noise distribution matrix	(n x m)
H	Linear (or linearized) observation matrix	(r x n)
I	Identity matrix	(various)
P	Covariance of state estimates	(n x n)
Φ	State transition matrix	(n x n)
Q	Covariance of input noise	(m x m)
W	Weighting matrix	(various)

Operator

$E[A]$	Mathematical expectation of A
A^T	The transpose of A
A^{-1}	The inverse of A
$A^\#$	The pseudo inverse of A
$A \equiv B$	A is equal to B by definition
$h_x(x^0)$	Indicates partial differentiation evaluated at x^0
$\ b\ _A^2$	Equivalent to $b^T A b$
$\text{tr}[A]$	The trace of A, equivalent to $\sum_i A_{ii}$

1. Introduction.

The problem to be considered is that of state estimation where the observations are nonlinear functions of the state of the system corrupted by additive white noises. The equations of motion of the state are linear in the state and the excitation which includes random components. A significant constraint on the solution is that the conceptual solution to this problem must lead to an explicit procedure which can be realized on a digital computer and the scheme must produce estimates as the observations are received.

The theory for the case where system dynamics and observation functions are linear is very highly developed. However, when nonlinearities are introduced there are virtually no completely satisfactory solutions.

Two methods for handling the nonlinear problem have been introduced. The first entails a linearization about a nominal trajectory in state space. Its success depends upon the accuracy of the nominal trajectory. This technique has little hope of success in a situation where there is almost no prior information about the trajectory. The target acquisition problem is an example of such a situation.

The second method is more of theoretical interest than as a candidate for a computational procedure. In this development the viewpoint is taken that the output of a state estimator should be the conditional probability density function, conditioned upon all past data. Computational difficulties arise from an effort to compute a complete function over the entire state space as compared to a more conventional estimator which selects a single point in the state space as the most likely state.

This study was particularly motivated by the difficulties encountered in filtering radar returns from an airborne target. The target dynamics

are presumably describable by a linear dynamic system where the elements of the state vector are the position and velocity of the target in cartesian coordinates measured from the radar. On the other hand, data available to a filter from the radar will usually be in spherical coordinates. Thus there exists a known, nonlinear relationship or transformation between the state of the system and observations. The results of a series of experiments using a filter based upon a simple linearization procedure are reported in Demetry and Hudson [4]. The operation of the filter was unsatisfactory under realistic conditions of initial uncertainty about target position, i.e., where to evaluate the partials involved in the linearization procedure.

The present work fills a gap in the field of nonlinear estimation for problem in which there is little prior information and a computationally feasible estimation procedure is required.

The problem is discussed and precisely formulated in Section 2. The original filtering problem is replaced by an associated minimization problem. The work of Kalman is reviewed since it is known that the Kalman filter is the solution to the associated minimization problem when the observations are linear functions of the states. A special single-stage form of the problem is considered. It is shown that the nonlinear observation problem can be approximated by a linear problem whose solution is known from Kalman's work. The resulting solution is then used to generate a new approximation to the original problem. Each iteration of the above process reduces the function to be minimized so long as the gradient is non-zero. It is shown that the whole class of problems originally considered can be cast in the special single-stage form.

The problem may be said to be solved at this point; however, for real-time calculations there must be procedures for controlling the number

of iterations. Iteration control procedures are discussed along with heuristic means for choosing the control parameters. Even with the number of iterations kept to a finite number, the computing requirements would increase indefinitely since each new set of data generates a new minimization problem over a larger number of variables. The control of this effect is discussed by introducing the concept of noise generated by the nonlinearities. The overall algorithm is discussed with respect to implementation on a digital computer.

The method was used on a realistic radar tracking problem. The models for the target and the radar are discussed. The results of the study indicate that the method produces reasonable estimates of the states of the system.

2. Detailed Statement of the Problem.

The specification of the problem may be viewed as consisting of four parts. Three of these are different types of information about the system under observation and the last is an implicit statement of how the available information should be combined to form an estimate of the state of the system.

The first type of information about the system is expressed by forming a dynamic model of the system. In this way the relation between states at different times is made explicit. It is only through the dynamic relation of the states that observations taken at diverse times have any relation to one another. In general usage the term filter is associated with a sequence of observations which are related to one another (correlated). The states and the dynamic model embody this correlation.

The second type of information is the relationship of the observation at a given time to the state at the same time.

The third type of information is the a priori knowledge about the state of the system.

Finally the best estimate of the state is defined. In general, none of the information about the state is definitive; it is all subject to some uncertainty and except for this uncertainty, it would be contradictory. The best estimate is defined to effect a certain compromise among all of the available information.

Since the resulting definition is implicit, computational difficulties occur. The resolution of these difficulties results in an explicit procedure for producing an estimate which is almost best within a reasonable computing time. Such an explicit procedure, or algorithm, will be

called a filter.

The Dynamic Model

The time evolution of the states of the system which is being observed are assumed to be adequately described by a difference equation which is linear in both the state and the excitation. The excitation is assumed to be a white random time sequence which has known first and second moments and is independent of the state. If the mean of the random excitation is not zero, then the random signal can be decomposed into a deterministic component and a zero-mean random component. The development will assume that there is no deterministic component (or non-zero mean). A short comment will be made at the appropriate point outlining the required changes for the case of deterministic inputs. These notions are concisely stated in (1) through (4).

$$x(k+1) = \Phi x(k) + \Gamma \omega(k) \quad (1)$$

$$E[\omega(k)] = 0 \quad (2)$$

$$E[\omega(k) \omega(j)^T] = \begin{cases} I & \text{for } k=j \\ 0 & \text{for } k \neq j \end{cases} \quad (3)$$

$$E[\omega(k) x(j)^T] = 0 \text{ for } k \leq j \quad (4)$$

where x is the state vector of n components,

Γ is a known $n \times r$ input distribution matrix,

ω is a random excitation of r components,

Φ is a known $n \times n$ state transition matrix,

$E[\]$ is the expectation operation, and

T denotes the transpose.

Equation (1) expresses the linearity of the state dynamics while (2), (3), and (4) express the qualities of zero mean, whiteness, and independence respectively about the excitation. The assumption that the covariance

of $\omega(k)$ is the identity matrix involves no loss of generality as long as Γ is appropriately chosen. Consider the two random excitations $\Gamma\omega$ and $\Gamma'\omega'$ where

$$\begin{aligned} E[\omega] &= 0 \\ E[\omega'] &= 0 \\ E[\omega \omega^T] &= I \\ E[\omega' \omega'^T] &= Q \end{aligned}$$

By comparing first and second moments, the two random excitations are equivalent if $\Gamma \Gamma^T = \Gamma' Q \Gamma'^T$. Q is a covariance matrix and thus is symmetric and positive semi-definite. This implies that a decomposition can be found such that $B B^T = Q$, and $\Gamma = \Gamma' B$.

The Observations

The data available to the filter are nonlinear functions of the state corrupted by additive white noise. The functional relation is assumed to be twice differentiable in the state. The corrupting noise is assumed to have zero mean and known variance. The noise is assumed to be independent of the states and the excitation. These notions are concisely expressed as

$$z(k) = h(x(k)) + v(k) \tag{5}$$

$$E[v(k)] = 0 \tag{6}$$

$$E[v(k) v(j)^T] = \begin{cases} R & \text{for } k=j \\ 0 & \text{for } k \neq j \end{cases} \tag{7}$$

$$E[v(k) x(j)^T] = 0 \tag{8}$$

$$E[v(k) \omega(j)^T] = 0 \tag{9}$$

where $z(k)$ is an m vector of observations at time k ,

$v(k)$ is an m vector of noise at time k ,

$h(\cdot)$ is an m vector of nonlinear functions of the states

R is the $m \times m$ covariance matrix of the measurement noise.

A Priori Information

In view of (1) any information relative to $x(k)$ must also be considered when estimating $x(k+1)$. In conventional linear sequential stage-by-stage estimation one can consider two distinct phases. The first is to bring forward all information from the past observations in the form of an a priori estimate using (1). The second phase is then to adjust this a priori estimate in view of the actual observations of the state. This is repeated from stage to stage. Clearly this process must start with a given a priori estimate for the first state. Sometimes this a priori estimate serves merely as a mathematical convenience for starting the filter [9]. In other cases an appropriate a priori estimate is really available. In any case an a priori estimate takes the form of an estimate of the initial state coupled with a measure of the accuracy of this estimate, the covariance of the error, defined as

$$E[(x(1)-x(1/0)) (x(1/0))^T] \equiv P(1/0) \quad (10)$$

$$E[x(1)-x(1/0)] = 0 \quad (11)$$

where $x(1/0)$ is the a priori estimate of $x(1)$, and

$P(1/0)$ is the covariance of the a priori estimate.

The double index argument will be used throughout to indicate an estimate of the state associated with the first index based on observations up to and including those associated with the second index.

Definition of Best Estimate

The best estimate of sequence of states $x(1)$ through $x(k)$ will be called $x(1/k)$ through $x(k/k)$ and is defined as that sequence which minimizes the scalar quantity

$$\begin{aligned}
C[x(1/k), x(2/k), \dots, x(k/k)] &\equiv \left\| x(1/k) - x(1/0) \right\|_{W_1}^2 \\
&+ \sum_{i=2}^k \left\| x(i/k) - \Phi x(i-1/k) \right\|_{W_2}^2 \\
&+ \sum_{i=1}^k \left\| z(i) - h(x(i/k)) \right\|_{W_3}^2
\end{aligned} \tag{12}$$

where the norm notation is introduced for compactness. By definition $\|b\|_A^2$ is equivalent to $b^T A b$ and the result is a scalar which is a quadratic function of the elements of b .

A best estimate defined in this manner might be called a weighted-least-squares estimate generalized to include the case where the quantity to be estimated is changing somewhat randomly in time.

The three different types of terms correspond to (10), (1), and (5) respectively. The terms inside the vertical bars are called residuals. The residuals are the difference between the expected value of a function of the true states and that same function evaluated at the estimate of the state.

This definition of the best estimate can be interpreted in several ways which will be developed below. These interpretations cannot in any sense prove that estimates defined in this way are best. The most that can be hoped for is that the interpretations offered will enhance the reasonableness of the resulting estimate. It must be realized that the best estimate is only what it is defined to be, which, in the final analysis is certainly somewhat arbitrary.

First, if all of the random sequences are assumed to be sequences of Gaussian (or normal) random variables, then it is possible to compute the probability of the observed data for a given sequence of states. This probability is viewed as a function of the true states. That sequence of

the states which maximizes the probability is taken as an estimate of the true states. The probability is commonly called the likelihood and can be expressed as

$$L(x(1), x(2), \dots, x(k)) = K \exp \left[-\frac{1}{2} C(x(1), x(2), \dots, x(k)) \right]$$

where K is a constant which is independent of the states if the weighing matrices are chosen as

$$W_1 = P(1/0)^{-1} \quad (13)$$

$$W_2 = (\Gamma \Gamma^T)^{-1} \quad (14)$$

$$W_3 = R^{-1} \quad (15)$$

The covariances are assumed to be nonsingular for simplicity. For a detailed discussion of the case of singular covariance matrix see Appendix I. It is clear that to maximize L one must minimize C . Thus for the case of Gaussian random variables the best estimate will be the so called maximum likelihood estimate.

A second point of view is that it would be desirable to find an estimate which resulted in zero residuals; requiring zero residuals, however, would imply a larger number of constraints than there are adjustable parameters (estimates). This suggests that the best estimate would be some sort of compromise where all of the residuals are small. Such a compromise is effected by setting up a weighted sum of squares of the residuals, C , as a function of the estimates, and selecting (or defining in this case) that set of estimates which minimize C as the best estimate.

In general, the best estimate will depend upon the weighting chosen for each residual. In order to determine the appropriate weighting matrices it is helpful to consider under what circumstances equal weighting would be appropriate. A heuristically reasonable answer might be to

weight equally when each of the random components have the same variance. If each of the sets of equations are multiplied by appropriate matrices new random variables can be defined so that each has unit variance. The residuals from these new equations are computed and their squares all weighted equally. In this form the residuals have an intuitively reasonable weighting. But it can be shown that this is equivalent to weighting the original residuals with the inverse of the corresponding covariance.

A very simple example should clarify the argument. Consider the problem of estimating x from two observations

$$z_1 = h_1(x) + v_1$$

$$z_2 = h_2(x) + v_2$$

where

$$E[v_1] = 0,$$

$$E[v_2] = 0,$$

$$E[v_1^2] = \sigma_1^2,$$

$$E[v_2^2] = \sigma_2^2.$$

Dividing the first equation by σ_1 and the second by σ_2 yields

$$z_1/\sigma_1 = h_1(x)/\sigma_1 + u_1$$

and

$$z_2/\sigma_2 = h_2(x)/\sigma_2 + u_2$$

where u_1 and u_2 are unit variance random variables. So the sum of squared residuals is formed

$$C(x) = (z_1/\sigma_1 - h_1(x)/\sigma_1)^2 + (z_2/\sigma_2 - h_2(x)/\sigma_2)^2$$

but this is equivalent to

$$C(x) = (z_1 - h_1(x))^2/\sigma_1^2 + (z_2 - h_2(x))^2/\sigma_2^2.$$

Therefore, if it is reasonable to weight equally residuals associated with random variables of equal variance then it follows for unequal variances the weighting should be in inverse proportion to the variance.

Another point of view is that the weighting matrices W_1 , W_2 and W_3 are chosen directly without recourse to any assumed random variables. There is, of course, an equivalent problem cast in terms of random variables. It is the author's view that it is easier to assess the magnitude of the variances of the random variables than to choose an appropriate set of weighting matrices directly.

A general model of the underlying physical process which generates the data has been presented. For any real physical situation the parameters Φ , Γ , R , $x(1/0)$ and $P(1/0)$ will be numerical quantities and the non-linear functions $h(x)$ will be a vector of explicit functions. This is the type of information which a filter designer must have before the filter can be constructed. For a given model there are many possible filters which might be considered; in each case presumably the output of the filter would be best in some sense, often unspecified. The filter under consideration in this work is defined by the sense in which the estimates are best, i.e., the filter is a least-squared-residuals filter. Thus, it should be noted that the filtering problem has been transformed into a sequence of minimization problems. It will turn out that the solution to each minimization problem is itself a sequence of solutions to a much simpler minimization problem, namely the problem with linear observation functions. The solution to this simpler problem is well known and is discussed in the next section along with other filter techniques where the model is similar to the one described in this section.

3. Prior Work.

The field of state estimation has received a great deal of interest recently, especially since the work of R. E. Kalman [7] and R. E. Kalman and R. Bucy [8] in linear estimation theory. For a comprehensive survey of the general field of estimation Deutsch [6] is suggested. Lee [9] presents a fine treatment of the theory, particularly with respect to the relationship between control and estimation theory. Cox [2, 3] is a specialized review of the efforts in the area of nonlinear estimation of which this work is a special case.

The structure of the problem and the results obtained by Kalman [7] along with several extensions, modifications and alternate interpretations are discussed in some detail. This discussion provides a convenient reference for comparison of the results of this thesis as well as an opportunity to establish certain known results which will be needed in the development of the method that follows. The method of trajectory linearization is discussed and it is shown how a natural extension leads to the method used here.

Kalman Filter

Kalman [7] has solved a special case of the problem under consideration where the measurements are linear functions of the state.

Symbolically

$$z(k) = H x(k) + v(k) \quad (16)$$

replaces (5).

Two cases are considered. In the first, all of the random sequences are assumed to be sequences of Gaussian random variables. With this assumption, the estimate is shown to be optimum in the sense that any linear function of the estimate is the minimum variance estimate of the same

linear function of the true state. The estimate turns out to be a linear function of the observations.

In the second case, there are no assumptions about the form of the probability density functions of the random variables, but the estimate is assumed to be a linear function of the observations. The optimum estimate is defined in the same way.

In either case the method of computing the optimum estimate (the filter) is the same. The sequence of operations can be envisioned as consisting of two steps. The first will be called the prediction equation.

$$x(k+1/k) = \Phi x(k/k) \quad (17)$$

The double argument notation will always indicate an estimate. The left side should be read; the estimate of the state at time $k+1$ given data up to time k . The second step will be called the adjustment for newly received data.

$$x(k+1/k+1) = x(k+1/k) + G(k) [z(k+1) - H x(k+1/k)] \quad (18)$$

where $[z(k+1) - H x(k+1/k)]$ is the error in the predicted observations, and $G(k)$ is a matrix of adjustment coefficients. The matrix $G(k)$ reflects the relative confidence one should have in the observed data as compared to the predicted estimate. This is discussed in [9].

$$G(k) = P(k+1/k) H^T [H P(k+1/k) H^T + R]^{-1} \quad (19)$$

Where $P(k+1/k)$ is the covariance of the estimates defined as follows:

$$P(k+1/k) = E[(x(k+1/k) - x(k+1))(x(k+1/k) - x(k+1))^T] \quad (20)$$

This formulation then requires that one must keep track of the covariance of the estimates. This is also done in two steps.

$$P(k+1/k) = \Phi P(k/k) \Phi^T + \Gamma \Gamma^T \quad (21)$$

$$P(k+1/k+1) = P(k+1/k) - P(k+1/k) H^T [H P(k+1/k) H^T + R]^{-1} H P(k+1/k) \quad (22)$$

Rauch [13] has derived these same equations based on the Gaussian assumption and shown the resulting estimate is the conditional mean and the maximum likelihood estimate.

Lee [9] has shown that the same equations yield a weighted least squared residual estimate where the weightings are the inverses of the covariance matrices. This also follows from the derivation of the maximum likelihood estimate based upon the assumption that the random sequences are Gaussian.

Trajectory Linearization

The trajectory linearization method has been used to solve nonlinear filtering problems such as orbit determination for artificial satellites [10], [11]. It is assumed from physical considerations that the evolution of the system state satisfies a general difference equation of the form

$$x^{(k+1)} = f(x(k), \omega(k), k) \quad (23)$$

and that observations are available in the form

$$z(k) = h(x(k), v(k), k) \quad (24)$$

where $\omega(k)$ and $v(k)$ are random vectors.

It is further assumed that there is a known nominal trajectory which is a sequence of states $x^0(k)$ such that

$$x^0(k+1) = f(x^0(k), 0, k) \quad (25)$$

It is desired to find the best estimate of the deviation of the true state from the nominal state.

Defining

$$y(k+1) \equiv x(k+1) - x^0(k+1) \quad (26)$$

and substituting (23) and (25) into (26) results in

$$y(k+1) = f(x(k), \omega(k), k) - f(x^0(k), 0, k) \quad (27)$$

This relation is now approximated by a first order Taylor series about

the point $x(k) = x^0(k)$ and $\omega(k) = 0$. The two partial derivatives are given appropriate symbols.

$$\Phi(k) \equiv f_x(x^0(k), 0, k) \quad (28)$$

$$\Gamma(k) \equiv f_\omega(x^0(k), 0, k) \quad (29)$$

The first order Taylor series expansion results in

$$\begin{aligned} y(k+1) \cong & f(x^0(k), 0, k) + \Phi(k) [x(k) - x^0(k)] \\ & + \Gamma(k) \omega(k) - f(x^0(k), 0, k) \end{aligned} \quad (30)$$

Substitution of (26) in (30) results in

$$y(k+1) \cong \Phi(k) y(k) + \Gamma(k) \omega(k) \quad (31)$$

From the nominal trajectory it is possible to construct a nominal set of observations

$$z^0(k) = h(x^0(k), 0, k) \quad (32)$$

Consider the deviations of the observations about the nominal observations.

$$u(k) \equiv z(k) - z^0(k) \quad (33)$$

Substituting (24) and (32) in (33) yields

$$u(k) = h(x(k), v(k), k) - h(x^0(k), 0, k) \quad (34)$$

The relation is approximated by a first order Taylor series about the point $x(k) = x^0(k)$ and $v(k) = 0$. The two partial derivatives are given appropriate symbols

$$H(k) \equiv h_x(x^0(k), 0, k) \quad (35)$$

$$S(k) \equiv h_v(x^0(k), 0, k) \quad (36)$$

$$\begin{aligned} u(k) \cong & h(x^0(k), 0, k) + H(k) [x(k) - x^0(k)] \\ & + S(k) v(k) - h(x^0(k), 0, k) \end{aligned} \quad (37)$$

Substitution of (26) in (37) yields

$$u(k) \cong H(k) y(k) + S(k) v(k) \quad (38)$$

Although it was not noted in the description of the Kalman filter, it is true that the equations remain valid if any or all of the matrices Φ , H , Γ , R , are known functions of time. These filter equations are thus directly applicable to (31) and (38).

The purpose of the process of trajectory linearization is to generate (31) and (38). It is then noted that with respect to the states $y(k)$ and the observations $u(k)$ the model is in the form of a linear dynamic system and linear observations. The Kalman filter is then applied directly as though (31) and (38) were equalities.

Nonlinear Noise

A question naturally arises concerning the adequacy of the first order approximation in developing (31) and (38). The heart of this problem is investigated by Denham and Pines [5] through the use of a very simplified model and a number of Monte Carlo studies. They reach the conclusion that the difficulties are of an indirect nature. The first estimates are about as good as might be expected. In processing subsequent data, however, trouble develops because the assumed quality of the first estimate is too great, which means that the next data get weighted too lightly.

This effect can be best seen by reconsidering (31). In order to make this expression into an equality, all of the higher order terms must be added to the right side of the expression. These additional terms should be considered as part of the observation noise. It is the failure to account for this nonlinear noise in the Kalman filter that causes the discrepancy between the covariance of the estimates as computed in the filter and the true average squared estimation errors. When the calculated covariance overstates the quality of a given estimate, a subsequent observation will certainly be combined with the current estimate in a non-

optimum manner.

Denham and Pines point out that when the order of magnitude of the expected value of the neglected terms is of the order of the natural measurement noise one cannot expect the linearized filter to work properly. The expression for the "nonlinear noise" involves the difference between the true state and the point about which the linearization takes place. If there were some way to reduce this difference then the nonlinear noise would be reduced correspondingly. These authors attribute to John Breakwell an iterative procedure for accomplishing this. The procedure is to linearize and filter, then relinearize at the new estimate and filter again. This cycle is repeated until the output of the filter is the same as the point at which the linearization takes place. A Monte Carlo study using this iterative technique revealed that the computed covariances quite accurately reflected the quality of the estimates.

It will be noted that the method used to solve the least-squared-residual problem as developed in the next section is exactly the iterative method suggested by Breakwell.

4. Development of the Solution Algorithm.

The development will proceed in two phases, the first being a conceptual means of finding the absolute best estimate, the second being the development of a series of compromises required for computational feasibility.

The minimization of (12) will have to be carried out in an iterative fashion since the simple process of differentiating and setting to zero does not lead to an explicit formula for the state estimates as it would if the observation functions were linear. The iterative procedure is based upon a linearization of the observation functions. The linearized observations are then in the form (16) and minimization is carried out using the Kalman filter equations. This produces a set of state estimates about which the nonlinear functions can be relinearized. This process is repeated until there is no further change in the state estimates.

The Kalman filter in its normal form is not completely adequate since its output is the sequence of estimates $x(1/1)$ through $x(k/k)$ while the point about which it is desired to linearize is $x(1/k)$ through $x(k/k)$. These latter estimates are called the smoothed estimates. There are formulas available, due to Rauch [13], for converting the output of the Kalman filter into smoothed estimates. There is, however, a more convenient way to get the same results in this case where all of the smoothed estimates are required. This involves converting from a multistage problem to a single-stage problem with a proportionately enlarged state space. The details of this conversion will be discussed following the discussion of the iteration for the single-stage process.

5. The Single Stage Minimization Procedure.

The equations related to the single-stage process are rewritten here using a simplified notation and are numbered using the same numbers as in their first appearance with an ' added.

$$z = h(x) + v \quad (5')$$

$$E[v] = 0 \quad (6')$$

$$E[vv^T] = R \quad (7')$$

$$E[vx^T] = 0 \quad (8')$$

$$E[(x-x_0)(x-x_0)^T] = P_0 \quad (10')$$

$$E[x-x_0] = 0 \quad (11')$$

$$C(x_1) = \|x_1 - x_0\|_{P_0}^2 + \|z - h(x_1)\|_R^2 \quad (12')$$

where

z is a vector of observations,

v is the noise in these observations,

R is covariance of the noise,

x_0 is the a priori estimate of the true state x ,

x_1 is the new estimate of x , and

P_0 is the covariance of the a priori estimate.

The pertinent equations from the solution to the linear problem are also rewritten here.

$$z = Hx + v \quad (16')$$

$$x_1 = x_0 + G(z - Hx_0) \quad (18')$$

$$G = P_0 H^T [H P_0 H^T + R]^{-1} \quad (19')$$

The non-singularity of P_0 assumed in (12') is fully discussed in Appendix I. It is sufficient here to say that if P_0 is singular the

problem can be reduced to a smaller state space where the associated P_0 is non-singular. It will always be assumed that R is non-singular, i.e., the assumption is made that there are no observations of unlimited accuracy.

The iterative minimization procedure involves a linear (first-order) approximation to (5') about a point x_1^i which will always be the best available estimate of x . This linearized approximation is then manipulated to form a synthetic observation which has the form (16'). This synthetic observation is used in (18') and a new approximation to the best estimate is obtained. Using this estimate as the point of linearization of (5') the process is repeated. These steps are expressed symbolically as follows.

$$z \cong h(x_1^i) + h_x(x_1^i) [x - x_1^i] + v \quad (39)$$

where x_1^i is the i th approximation to the best estimate x .

$$z^i \equiv z - h(x_1^i) + H^i x_1^i \quad (40)$$

$$z^i \cong H^i x + v \quad (41)$$

where z^i is the so-called synthetic observation and

$$H^i = h_x(x_1^i) \quad (42)$$

$$x_1^{i+1} = x_0 + G^i (z^i - H^i x_0) \quad (43)$$

where

$$G^i = P_0 H^{iT} [H^i P_0 H^{iT} + R]^{-1} \quad (44)$$

This completes the description of the pure minimization algorithm except for a specification of the initial point of linearization and the possibility of overshoot. Discussion on the initial point of linearization will follow after the discussion of conversion from multi-stage to single

stage. The possibility of an overshoot results from the fact that a simple first order approximation to the nonlinearity may not be accurate for the subsequent change in x_1 . The overshoot protection scheme used is simply if

$$C(x_1^{i+1}) \geq C(x_1^i)$$

then x_1^{i+1} is replaced by $1/2 (x_1^{i+1} + x_1^i)$,

It will now be shown that each step in this process does result in a decrease in the cost, C . Since if the new cost is greater than the old cost the step size, $x_1^{i+1} - x_1^i$, is reduced by a factor of 2 it only is necessary to show that for a small enough step size there will be a reduction in C . The demonstration will be begun by showing that the change in estimate is related to the negative gradient of the cost evaluated at x_1^i by a positive definite matrix.

$$g(x_1^i) \equiv -\frac{1}{2} C_{x_1} (x_1^i) \quad (45)$$

$$g(x_1^i) = P_o^{-1} (x_o - x_1^i) + H^{iT} R^{-1} [z - h(x_1^i)] \quad (46)$$

The change in the iterate is

$$d \equiv x_1^{i+1} - x_1^i \quad (47)$$

$$d = x_o - x_1^i + G^i [z - h(x_1^i)] \quad (48)$$

$$d = x_o - x_1^i + G^i [z - h(x_1^i) + H^i x_1^i - H^i x_o] \quad (49)$$

$$d = [I - G^i H^i] (x_o - x_1^i) + G^i [z - h(x_1^i)] \quad (50)$$

Now to show that

$$d = [P_o - P_o H^{iT} [H^i P_o H^{iT} + R]^{-1} H^i P_o] g(x_1^i) \quad (51)$$

it must be shown that

$$[I - G^i H^i] = [P_o - P_o H^{iT} [H^i P_o H^{iT} + R]^{-1} H^i P_o] P_o^{-1} \quad (52)$$

and that

$$G^i = [P_o - P_o H^{iT} [H^i P_o H^{iT} + R]^{-1} H^i P_o] H^{iT} R^{-1} \quad (53)$$

The first of these is obvious after substituting for G^i .

In the latter $P_o H^{iT}$ is factored from the left side to obtain

$$P_o H^{iT} [I - [H^i P_o H^{iT} + R]^{-1} H^i P_o] R^{-1}$$

and then $[H^i P_o H^{iT} + R]^{-1} [H^i P_o H^{iT} + R]$ is substituted for the I above.

$$P_o H^{iT} [H^i P_o H^{iT} + R]^{-1} [H^i P_o H^{iT} + R - H^i P_o H^{iT}] R^{-1}$$

After cancellation, the above is the expression for G^i .

Note the matrix $P_o - P_o H^{iT} [H^i P_o H^{iT} + R]^{-1} H^i P_o$ is the covariance of the updated estimate in the linear case so it will be given the special symbol

$$P_1 \equiv P_o - P_o H^{iT} [H^i P_o H^{iT} + R]^{-1} H^i P_o \quad (54)$$

so that (51) can be written

$$d = P_1 g(x_1^i) \quad (55)$$

After applying the overshoot control the actual change of the iterate is in the direction of d but may have a smaller magnitude. Let D be the actual change so that

$$D = qd \quad (56)$$

$$D = q^p P_1 g(x_1^i) \quad (57)$$

where q is some integral power of $(1/2)$.

Then to a first-order approximation the change in C , ΔC , is given by

$$\Delta C \approx 2g^T D \quad (58)$$

$$\Delta C \approx 2q g^T d \quad (59)$$

$$\Delta C \approx 2q \mathbf{g}^T P_1 \mathbf{g} \quad (60)$$

$$\Delta C \approx 2q \|\mathbf{g}\|_{P_1}^2 \quad (61)$$

Since all higher order terms in the expansion of ΔC involve higher powers of q it can be asserted that for some small enough q the first order term will dominate. Thus for some suitable q the change in C will be negative for any non-zero \mathbf{g} if P_1 is a positive definite matrix. P_1 is known from the linear theory to be positive definite for any value of H^i and any positive definite R . This also follows from the fact that

$$P_1^{-1} = P_0^{-1} + H^i T R^{-1} H^i \quad (62)$$

a convenient matrix identity discussed in [1], and the fact that the inverse of a positive definite matrix is positive definite.

It has been shown that the special single-stage problem can be solved by solving a sequence of simple problems which approximate more and more closely the real problem. Each iteration was shown to reduce the cost unless the gradient of the cost was zero.

In the next section it will be shown that the general problem considered in this method can be recast in the form of a single-stage problem.

6. Multi-stage Case Cast in Single stage Form.

The process of converting a multi-stage estimation problem into a single stage problem is accomplished by defining the new state to be the juxtaposition of the states at the various stages. The process will be carried out in detail for a two-stage problem and then the process will be generalized by induction for a k-stage process.

Let the new state be

$$x \equiv \begin{bmatrix} x(1) \\ \text{-----} \\ x(2) \end{bmatrix}, \quad (63)$$

let the new estimate be

$$x_1 \equiv \begin{bmatrix} x(1/2) \\ \text{-----} \\ x(2/2) \end{bmatrix}, \quad (64)$$

let the a priori for the new state be

$$x_0 \equiv \begin{bmatrix} x(1/0) \\ \text{-----} \\ x(2/0) \end{bmatrix}, \quad (65)$$

and let the a priori covariance of the new state be

$$P_0 \equiv \begin{bmatrix} P(1,1/0) & | & P(1,2/0) \\ \text{-----} & | & \text{-----} \\ P^T(1,2/0) & | & P(2,2/0) \end{bmatrix}. \quad (66)$$

Some of the elements in x_0 and P_0 have not been previously defined. However, the notation used has already been defined, i.e., $x(2/0)$ means the estimate of the second state given no data. The submatrices in P_0 are defined as follows:

$$P(i,j/0) \equiv E[(x(i/0) - x(i))(x(j/0) - x(j))^T] \quad (67)$$

This is a natural extension of the double argument notation already defined which is necessary to accommodate consideration of the cross correlation between states at various times. Since these new elements are not given directly in the multi-stage model it will be necessary to fill in these

elements in order to define the single-stage model.

First consider $x(2/0)$. Based upon the requirement that (11) be satisfied it must be true that

$$x(2/0) = E[x(2)] \quad . \quad (68)$$

Substituting for $x(2)$ from (1) and taking advantage of (2), the fact that $\omega(1)$ has mean zero, yields

$$x(2/0) = \Phi E[x(1)] \quad . \quad (69)$$

Introducing (11) above yields

$$x(2/0) = \Phi x(1/0) \quad . \quad (70)$$

If there are deterministic inputs (or equivalently $E[\omega(k)] \neq 0$) the appropriate modification is

$$x(2/0) = \Phi x(1/0) + \Gamma E[\omega(1)] \quad . \quad (71)$$

Now consider the various submatrices of P_0 . $P(1,1/0)$ is already known in the multi-stage problem as $P(1/0)$. $P(1,2/0)$ is given by

$$P(1,2/0) = E[(x(1/0) - x(1))(x(2/0) - x(2))^T] \quad . \quad (72)$$

Substituting (70) and (1) in the last part of (72) yields

$$P(1,2/0) = E[(x(1/0) - x(1))(\Phi x(1/0) - \Phi x(1) - \Gamma \omega(1))^T] \quad . \quad (73)$$

Taking advantage of the independence of $\omega(1)$ from (2) and factoring Φ^T yields

$$P(1,2/0) = E[(x(1/0) - x(1))(x(1/0) - x(1))^T] \Phi^T \quad (74)$$

but this is just

$$P(1,2/0) = P(1/0) \Phi^T \quad . \quad (75)$$

By similar arguments it can be shown that

$$P(2,1/0) = \Phi P(1/0) \quad (76)$$

and

$$P(2,2/0) = \Phi P(1/0) \Phi^T + \Gamma \Gamma^T \quad . \quad (78)$$

The remaining elements needed to complete the description of the single-

stage model have to do with the observation process.

Let the actual observations be

$$z \equiv \begin{bmatrix} z(1) \\ \text{---} \\ z(2) \end{bmatrix}, \quad (79)$$

let the vector of observation functions be

$$h(x) \equiv \begin{bmatrix} h(x(1)) \\ \text{---} \\ h(x(2)) \end{bmatrix}, \quad (80)$$

let the measurement noise be

$$v \equiv \begin{bmatrix} v(1) \\ \text{---} \\ v(2) \end{bmatrix}, \quad (81)$$

and let the measurement noise covariance be

$$R \equiv \begin{bmatrix} R & | & 0 \\ \text{---} & & \text{---} \\ 0 & | & R \end{bmatrix} \quad (82)$$

where the off-diagonal elements of R are zero matrices in view of the whiteness of the observation noise.

As the number of stages is increased, the dimension of the single-stage x is also increased. The process of expanding the single-stage model of a k stage model to that of a $(k + 1)$ -stage model is explained in detail only for x , x_0 , and P_0 . The expansion process or augmentation for the remaining elements of the model is as simple as the augmentation of x will prove to be. The augmentation of x is shown explicitly as an example.

In the augmentation process x goes from

$$x = \begin{bmatrix} x(1) \\ \text{---} \\ \cdot \\ \cdot \\ \cdot \\ \text{---} \\ x(k) \end{bmatrix} \quad \text{to} \quad \begin{bmatrix} x(1) \\ \text{---} \\ \cdot \\ \cdot \\ \cdot \\ \text{---} \\ x(k) \\ \text{---} \\ x(k+1) \end{bmatrix}$$

The expansion of x_0 is only slightly more involved. For the $(k + 1)$ -stage case the a priori estimate is

$$x_0 = \begin{bmatrix} x(1/0) \\ - - - \\ \vdots \\ \vdots \\ x(k/0) \\ - - - \\ x(k + 1/0) \end{bmatrix} \quad (83)$$

where $x(k + 1/0) = \Phi x(k/0)$.

The structure of the P_0 corresponding to the $(k + 1)$ -stage case is a $(k + 1)$ by $(k + 1)$ square matrix whose elements are submatrices. The upper left k by k part of this matrix is already known from the P_0 associated with the k -stage model. Thus to expand to the $k + 1$ case it is only necessary to fill in the lower border. The a priori covariance P_0 has the form

$$P_0 = \begin{bmatrix} P(1,1/0) & . & . & . & P(1,k/0) & \vdots & P(1,k + 1/0) \\ \vdots & & & & \vdots & \vdots & \vdots \\ \vdots & & & & \vdots & \vdots & \vdots \\ P(k,1/0) & . & . & . & P(k,k/0) & \vdots & P(k,k + 1/0) \\ \hline P(k + 1,1/0) & . & . & . & P(k + 1,k/0) & \vdots & P(k + 1, k + 1/0) \end{bmatrix} \quad (84)$$

The formulas below for the new border elements of P_0 were derived in exactly the same way the submatrices $P(1,2/0)$, $P(2,1/0)$ and $P(2,2/0)$ were found in the case for $k + 1 = 2$.

$$P(k + 1,i/0) = \Phi P(k,i/0) \quad \text{for } 1 \leq i \leq k \quad (85)$$

$$P(i,k + 1/0) = P^T(k + 1,i/0) \quad \text{for } 1 \leq i \leq k \quad (86)$$

$$P(k + 1,k + 1/0) = \Phi P(k,k/0) \Phi^T + \Gamma \Gamma^T \quad (87)$$

Thus it is clear that the dynamic relations represented by (1) for the multi-stage problem are incorporated into the very special structure of the large covariance matrix P_0 in the single-stage equivalent.

After a given multi-stage problem has been cast in single-stage form,

the single-stage problem is solved using the methods previously described. The outcome of the single-stage solution is the best estimate, x_1 , which must then be interpreted in terms of the original multi-stage problem. The best estimate, x_1 , is the best estimate of all states given all data up to the present stage, i.e.

$$x_1 = \begin{bmatrix} x(1/k) \\ - \\ - \\ - \\ \cdot \\ \cdot \\ \cdot \\ - \\ - \\ - \\ x(k/k) \end{bmatrix} \quad (88)$$

Finally, as each new single-stage solution process is started there must be an initial estimate of x_1 which is called x_1^1 . The first iterate for a $(k + 1)$ -stage minimization will be based on the final estimate from the previous minimization over k stages. Explicitly, the first iterate is given by

$$x_1^1 = \begin{bmatrix} x(1/k) \\ - \\ - \\ - \\ \cdot \\ \cdot \\ \cdot \\ - \\ - \\ - \\ x(k/k) \\ - \\ - \\ - \\ x(k + 1/k) \end{bmatrix} \quad (89)$$

where $x(k + 1/k) = \Phi x(k/k)$ (90)

At this stage in the development of the filter, the problem has been solved but in a very impractical way. The filter has been broken into an unlimited sequence of minimization problems. Each minimization problem is solved through an as yet unlimited sequence of approximate solutions. In order to design a practical filter a realistic convergence test must be used to terminate the minimization process. Such a convergence test is discussed in the next section. Another difficulty arises in connection

with the sequence of minimization problems. As more and more data are considered, spanning a larger and larger collection of states, the size of the minimization problem increases. A practical means of limiting the size (dimensionality) of the minimization problem will be discussed in a subsequent section.

7. Criteria for Termination of the Minimization Procedure.

It seems to be a universal feature of any iterative numerical method that the termination decision is based upon "feel" or "rules of thumb". Typically, a quantity is chosen as a measure of the convergence of the iterative process and this measure is compared against a standard or threshold. Fortunately, in this particular case it is possible to offer some insight into the choice of both the measure and the standard or threshold. This is true because the problem is basically a stochastic one. By analogy with certain special cases it is possible to approximate the probability density function of the measure of interest.

The control law or algorithm for the control of the number of iterations is based upon the fact that the minimum cost, $C(x_1)$, is itself a random variable whose distribution function can be approximated by that of a chi square random variable. The minimum cost is exactly distributed as a chi square variable when the measurements are linearly related to the states and all of the random variables are Gaussian. The chi square distribution is characterized by a parameter called the number of degrees of freedom. For the least square problem the number of degrees of freedom is the number of constraint equations less the number of parameters adjusted in the process of minimizing the sum of the squares.

Using this assumed distribution function for the minimum value of C it is possible to evaluate numerically the probability of a minimum C being greater than some number C_L . Actually the question is reversed so that a C_L is found such that the probability that a minimum C is greater than C_L is some small number α . This number $C_L(\alpha)$ is then used as a threshold for comparison with the actual value of C after each iteration. If C is greater than C_L the process is reiterated on the assumption that it is

very improbable that this value of C is in fact a minimum.

Clearly using such a test will reject a certain number of true minimum values of C . For this small percentage (α) of cases, the stopping criteria is based upon the relative change in state estimates. When the relative change in each component of the state vector after an iteration is less than some small number, EPS , the successive estimates are considered to be equal and the process is terminated.

On the other hand, passing this first test does not assure that a minimum has been reached. For this reason, a second test is prescribed which specifies a minimum improvement. Choosing the minimum improvement, C_M , may be considered purely arbitrary. On the other hand it may be helpful to invoke a statistical interpretation to aid in choosing C_M . Such an interpretation exists if all of the random sequences are Gaussian. Then $C(x^i)$ is related to the likelihood of x^i and $C(x^i) - C(x^{i+1})$ is related to the likelihood ratio. The likelihood ratio is a common statistic used to test the significance of the difference between two estimates.

The difference is considered to be significant at the β level if the probability of occurrence of the observed likelihood ratios is less than β under the assumption that x^i is the true state. The test of statistical significance is made by setting a threshold on the likelihood ratio, or some function of it. The difference, $C(x^i) - C(x^{i+1})$, is minus twice the likelihood ratio and has a chi square distribution with the number of degrees of freedom equal to the number of components in the state, x . C_L is chosen as that value for which the probability of a chi square variable less than C_L is β . The test is: if $C(x^i) - C(x^{i+1})$ is greater than C_L re-iterate, otherwise terminate the procedure. The satisfaction of the test suggests the interpretation that the last two estimates are not significantly

different so no further iteration is carried out.

The convergence test described may be summarized in terms of three quantities involved in the minimization process and three thresholds. The three quantities are 1) r , the maximum absolute relative change in any component of the state vector from one iteration to the next, 2) C^i , the current value of the cost and 3) $(C^{i-1} - C^i)$, the change in the cost over the last iteration. The corresponding three threshold parameters are EPS , C_M , C_L . The decision rules for terminating or continuing the process are displayed in Figure 1.

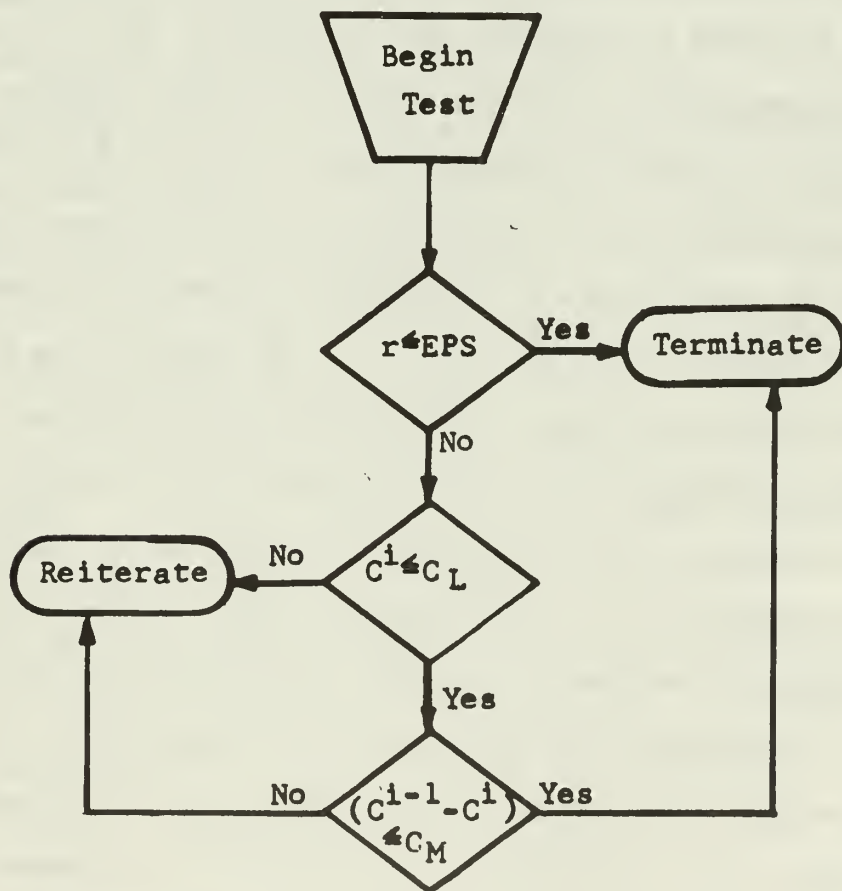


Figure 1. Flow graph of the criteria for iteration termination.

As an example, consider a single stage minimization where the state has six components and the measurement has four components. The probability that a chi square variable having four degrees of freedom will have a value greater than 14.9 is 0.005. This suggests that if $C_L \cong 14.9$ only one true minimum out of 200 will be rejected by this test. The probability that a chi square variable with six degrees of freedom will be less than 0.872 is 0.01. If $C(x^i) - C(x^{i-1}) \leq C_M \cong .872$ the last two iterations are not significantly different at a 0.99 confidence level. Finally, for those unusual cases where the true minimum is greater than 14.9 the iterations are continued until the iteration values are the same within the limitations of computer word length. For the CDC 1604 the floating operations carry about ten significant digits. It should be considered that absolute convergence has been attained when the relative change in all of the components of the estimate do not change more than one part in 10^9 . This indicates a choice for EPS of 10^{-9} .

The remarks of this section were directed toward providing some insight into the choice of the threshold parameters. While the assumptions which would make these interpretations rigorous may in most cases be lacking, the filter designer must incorporate a convergence test, i.e., he must choose a set of parameters. The interpretations discussed above are offered as an aid toward choosing an efficient set of parameters.

8. Criteria for the Number of Smoothing Stages to be Carried.

As a result of the way in which the best estimate has been defined, the complexity of the algorithm increases with the number of stages over which data is available. The estimate of state at the initial time is the result of a minimization involving n (number of components in the state vector) variables. The estimate of the state at the time of the twelfth measurement would be the result of a minimization over $12n$ variables. Clearly, proceeding in this way the computational requirements will exceed the capabilities of any computer after a finite number of stages.

The work of Denham and Pines [5] has focused attention on the difference between the case where the measurements are linear in the states and the more general nonlinear case. This difference was shown to be the result of neglecting higher-order terms in the expansion of the measurement function. The minimization over many stages may be viewed as a means of avoiding this problem since each linearization is only tentative. As new data become available, providing more information about the old states, the linearization of the measurement functions becomes more accurate. When the state is known well enough so that the second-order terms are negligible, the linearized measurement is considered to be accurate. This approach will be developed into a criterion for the number of smoothing stages which must be carried in the next minimization process.

Consider a second-order expansion of a single nonlinear observation function about a point x^0 .

$$z \cong h(x^0) + h_x(x^0)(x-x^0) + \frac{1}{2}(x-x^0)^T h_{xx}(x^0)(x-x^0) + v \quad (91)$$

where $h_x(x^0)$ is a row vector of partial derivatives of h evaluated at x^0 and $h_{xx}(x^0)$ is the symmetric matrix of second partials of h evaluated at x^0 .

In order to form a synthetic observation, z^0 , of the form of (16), the synthetic observation must be a linear function of the true state with added noise which has zero mean.

$$z^0 \equiv z - h(x^0) + h_x(x^0) x^0 - b \quad (92)$$

$$z^0 \approx H x + v + v' \quad (93)$$

where $H \equiv h_x(x^0)$ and

$$b \equiv \frac{1}{2} E[(x-x^0)^T h_{xxx}(x^0) (x-x^0)] \quad (94)$$

and v' is the variation of the second-order term about its mean, b .

The added noise v' is considered to be the random noise caused by the linearizing process. The magnitude of this nonlinear noise can be measured in terms of its variance, R' . Expressions for evaluating b and R' are developed in Appendix II.

The process of dropping a stage of smoothing is a lumping operation. It can be seen by examining the equations for the linear filter that all past data is brought forward in time through (17) and (21). This is not done immediately in the nonlinear filter because the observation equations have been linearized at a point which may be quite different from the true state. By keeping several stages active in the filter it is possible to perform the linearization at a point much closer to the true state. The dropping of a stage should be accompanied by a high degree of confidence that the last linearization was performed at a point close to the true state. That is, H is not going to change significantly as better estimates of the true state become available. The invariance of H is related to the expression for $R' = \frac{1}{2} \text{trace}[h_{xxx}(x^0)P]$ since $h_{xxx}(x^0)$ represents the variability of H with x and P represents the variance of x^0 . Thus when the natural noise in each element of the observation vector for a given

stage is an order of magnitude greater than the nonlinear noise, the stage corresponding to observation no longer carried.

The mechanics of lumping are quite straight forward once it has been decided that the current linearization is a good approximation. Under these conditions, the Kalman sequential filter equations are directly applicable. If these equations are applied to those stages which are to be lumped, the result will be an a priori estimate of the first state which is not to be lumped. The states which have been lumped no longer appear. All of the information that these states carried with respect to the estimation of future states is characterized by the a priori estimate of the first state which is not lumped. From this point then, the problem has exactly the same structure as the problem before lumping except that there are fewer stages being carried.

9. Computation of the Solution.

The basic features of the algorithm are shown in Figure 2. The following is a brief resume of the quantities which must be computed in order to implement the filter. Consider the overshoot decision. In order to decide whether an overshoot has occurred it will be necessary to evaluate the cost, C . For this purpose the multi-stage expression (12) is more convenient than the single stage expression (12'). To test for convergence it is necessary to have the current cost, the previous cost, the current estimate, the previous estimate and the two parameters C_L and C_M which are functions of the number of smoothing stages currently being carried. The decision on the number of stages to be carried depends upon an evaluation of the nonlinear noise associated with each observation. In order to evaluate this nonlinear noise, the matrix of second partial derivatives of the observation functions must be computed at the current best estimate of the true state. In addition, there must be available a covariance matrix representing the uncertainty of this estimate.

Computation of the Cost

In general there are many means of computing a given quantity. It turns out that the expressions for some quantities are useful in discussing the problem but are not the most efficient in actual computation. Such is the case with the cost C . For discussion purposes the cost was expressed in terms of the single-stage state variable. For computing the cost at an actual estimate, however, the form of (12) is more efficient.

The first term is handled as follows:

$$\|x(1/0) - x(1/k)\|_{W_1}^2 = \|B(x(1/0) - x(1/k))\|_I^2 \quad (95)$$

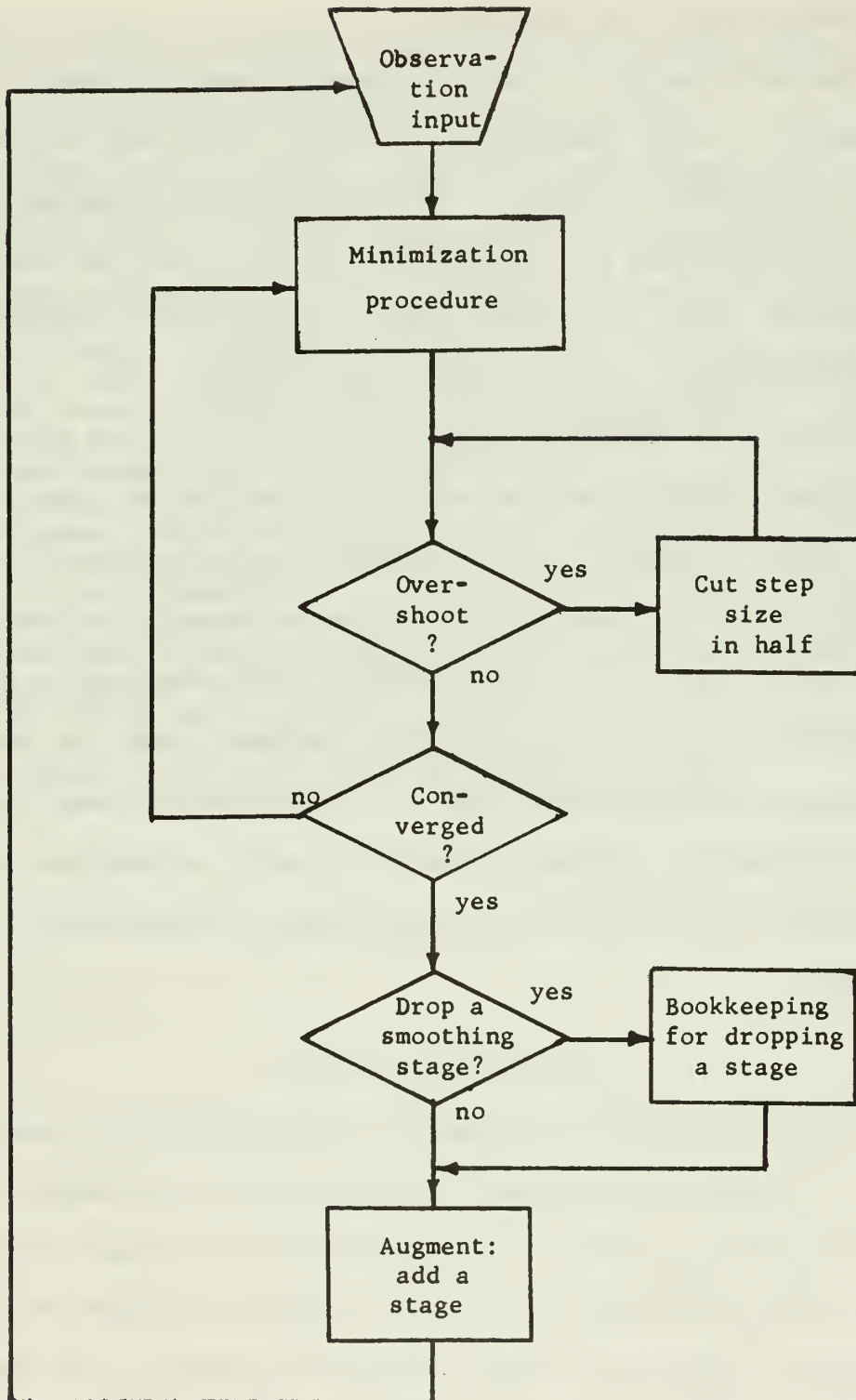


Figure 2. Flow graph of overall procedure.

where $W_1 = P^\#(1/0)$ (the pseudo inverse of $P(1/0)$) and B is chosen so that the equality holds which implies that B satisfies

$$B^T B = P^\#(1/0) \quad . \quad (96)$$

A suitable B is found by a special routine which begins by decomposing $P(1/0)$ into the form

$$P(1/0) = AA^T \quad (97)$$

where A is an $n \times r$ matrix and r is the rank of $P(1/0)$. If $r = n$ then

$$P^\#(1/0) = P^{-1}(1/0) \quad \text{and} \quad (98)$$

$$B = A^{-1} \quad . \quad (99)$$

If $r \leq n$ then

$$P^\#(1/0) = A^T \# A^\# \quad (100)$$

which implies that $B = A^\#$ and $A^\#$ can be computed by $A^\# = (A^T A)^{-1} A^T$, where the indicated inverse is known to exist by construction. That is, $A^T A$ is $r \times r$ and has rank r so it is non-singular. The routine which computes A also computes A^{-1} if it exists, with only minor additional labor.

For most applications $P(1/0)$ will not be singular even though the single-stage covariance will be singular if Γ has rank less than the system order. It is for this reason that the procedure adopted has a built-in flexibility to handle the singular case but handles the non-singular case with virtually no loss of computational efficiency over the more conventional approach of inverting directly the covariance matrix $P(1/0)$.

Evaluating the typical second term in (12) is accomplished in a similar fashion

$$\|x(i/k) - \Phi x(i-1/k)\|_{W_2}^2 = \|S_1 x(i/k) - S_2 x(i-1/k)\|_I^2 \quad (101)$$

where $W_2 = (\Gamma \Gamma^T)^\#$. Since Γ and Φ are assumed to be constant throughout

the problem S_1 and S_2 can be computed beforehand. By comparing terms

$$S_1^T S_1 = (\Gamma \Gamma^T)^{\#} \quad (102)$$

$$\text{and } S_2 = S_1 \Phi \quad (103)$$

There is no loss of generality in assuming that Γ has full rank, i.e., rank equal to its minimum dimension. Under these conditions

$$(\Gamma \Gamma^T)^{\#} = \Gamma^{\#T} \Gamma^{\#} \quad (104)$$

where $\Gamma^{\#} = (\Gamma^T \Gamma)^{-1} \Gamma^T$. This implies that

$$S_1 = (\Gamma^T \Gamma)^{-1} \Gamma^T \quad (105)$$

$$\text{and } S_2 = (\Gamma^T \Gamma)^{-1} \Gamma^T \Phi \quad (106)$$

where the indicated inverse is known to exist.

Finally the third typical term of (12) is

$$\|z(k) - h(x(1/k))\|_{W_3}^2 \quad (107)$$

where

$$W_3 = R^{-1}$$

The procedure used to evaluate this term assumes that the measurement errors are independent, i.e., R is diagonal. While this is a realistic assumption for most real problems there are means similar to those already used to handle cases where the observation errors at any time are correlated with one another, i.e., R is not diagonal. The details will not be discussed for lack of physical motivation.

This computation is carried out in the computer subroutine COST. The auxiliary matrices B , S_1 and S_2 are computed outside the subroutine. The matrix decomposition routine which generates A such that $AA^T = P(1/0)$ is called DECOMA.

The Basic Minimization Procedure

Another procedure which is computed by a different method than that used in the development is the minimization process (43). The computational procedure is a step-by-step solution of the linearized single-stage problem. This single-stage problem is viewed as involving a sequence of observations. Each observation is combined in turn to produce a new estimate of the state and a new covariance of that state. Figure 3 shows the details of the step-by-step procedure. At any point in the procedure the best estimate, given the data processed, is \hat{x} and the associated covariance is P . This type of sequential processing is valid under the assumption that the observation errors are mutually independent. In the computer program this process is carried out with the observations divided into blocks according to the time of the observation. The subroutine KALFIL processes each block in the manner indicated by Figure 3.

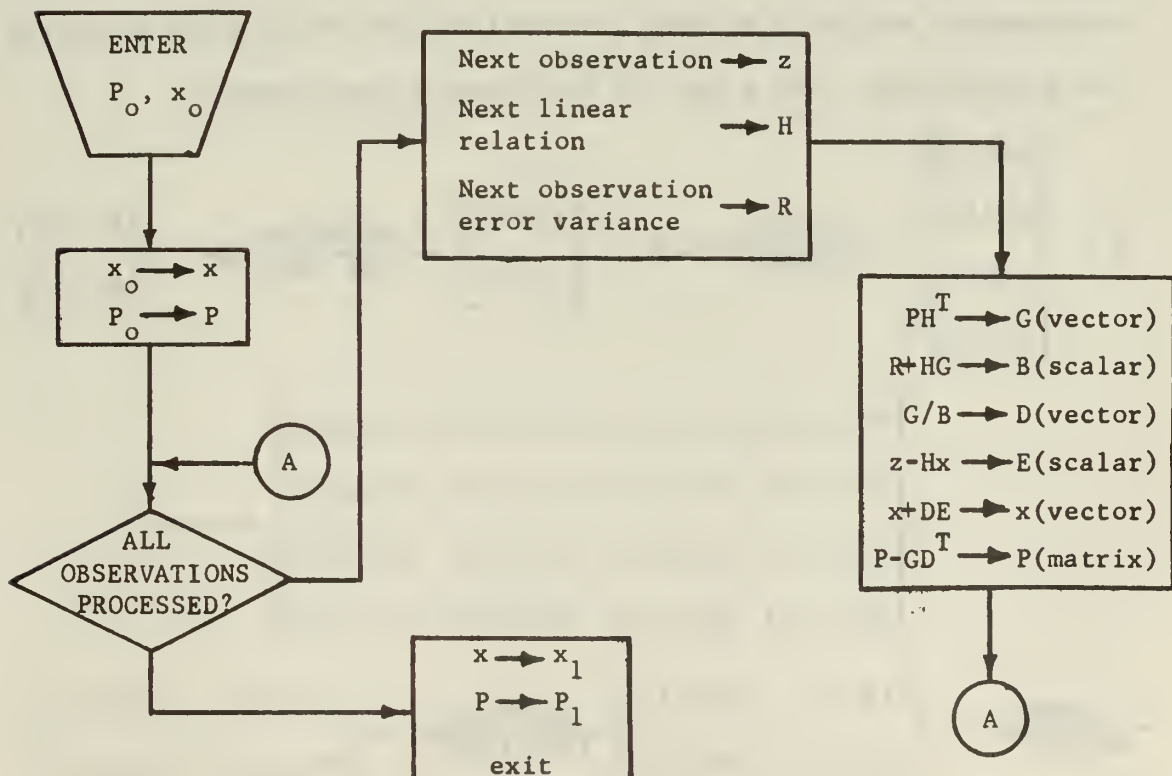


Figure 3. Flow graph of step-by-step minimization procedure.

This routine is entered once for each block or once for each smoothing stage carried.

This process yields the useful result that the vector x after j blocks is composed of the sequences of state estimates $x(1/j)$, $x(2/j)$, ..., $x(k/j)$. Similarly the matrix P is composed of k^2 submatrices of the form

$$P = \begin{bmatrix} P(1,1/j) & P(1,2/j) & . & . & P(1,k/j) \\ P(2,1/j) & . & . & . & . & . \\ P(k,1/j) & . & . & . & . & P(k,k/j) \end{bmatrix}$$

It is at this point that a lumping operation takes place. If it has been decided that all of the observations up to and including the j^{th} stage have negligible nonlinear noise, then a lumping operation is performed. This consists of shifting all of the estimates x up and out and shifting the P matrix up and to the left. This has the effect of eliminating any reference to any state at time j or earlier and reducing the dimension of the single-stage state x and the single-stage covariance P .

$$x = \begin{bmatrix} x(1/2) \\ x(2/2) \\ x(3/2) \\ x(4/2) \end{bmatrix} \xrightarrow[\text{SHIFTING}]{\text{AFTER}} x = \begin{bmatrix} x(3/2) \\ x(4/2) \end{bmatrix} \xrightarrow[\text{TIME INDEX}]{\text{SHIFTING}} x = \begin{bmatrix} x(1/0) \\ x(2/0) \end{bmatrix}$$

$$P = \begin{bmatrix} P(1,1/2) & P(1,2/2) & P(1,3/2) & P(1,4/2) \\ P(2,1/2) & P(2,2/2) & P(2,3/2) & P(2,4/2) \\ P(3,1/2) & P(3,2/2) & P(3,3/2) & P(3,4/2) \\ P(4,1/2) & P(4,2/2) & P(4,3/2) & P(4,4/2) \end{bmatrix} \longrightarrow$$

$$\xrightarrow[\text{SHIFTING}]{\text{AFTER}} P = \begin{bmatrix} P(3,3/2) & P(3,4/2) \\ P(4,3/2) & P(4,4/2) \end{bmatrix} \xrightarrow[\text{TIME INDEX}]{\text{SHIFTING}} P = \begin{bmatrix} P(1,1/0) & P(1,2/0) \\ P(2,1/0) & P(2,2/0) \end{bmatrix}$$

Figure 4 Evolution of the estimate and covariance during the transition

The process of shifting in the computer produces an automatic change in indices. The process is shown as two step for an example where $j = 2$, $k = 4$ in Figure 4. The subroutine SHIFT performs the details of shifting the matrices as indicated above as well as several other matrices which must be shifted.

10. A Simple Example.

Unfortunately examples seem to fall in two mutually exclusive categories, enlightening or realistic. The following example is introduced to illustrate the mechanical details of the algorithm. This example also illustrates the process of abstracting the mathematical model from the physical situation. Consider an active, drunken, tight rope walker. He is put on a tight rope so that only one coordinate will be needed to specify his position which will be designated $x(k)$. A drunken person is considered in order to introduce the concept that his position is a random quantity.

It will be assumed from previous experience with drunken tight rope walkers that his next position is different from his last position by some completely random variable. The mean squared value of this difference is assumed to be known. Further it is assumed that his ramblings to the right are balanced in the long run by those to the left, i.e., they have zero mean. The mathematical model for this much of the physical situation is given below.

$$x(k+1) = x(k) + \omega(k) \quad (108)$$

$$E[\omega(k)] = 0 \quad (109)$$

$$E[\omega(k)\omega(j)] = \begin{cases} Q & \text{for } j=k \\ 0 & \text{for } j \neq k \end{cases} \quad (110)$$

For concreteness Q is taken as 0.02.

The first expression is usually said to be the model of a dynamic system excited by white noise. The second and third are quantitative descriptions of the noise.

The next part of the situation that must be described is the process by which data are obtained. Here the concept of a nonlinear measurement is introduced. An angular measurement is made at some fixed distance

from the tight rope, i.e., the observer turns his head through a certain angle. For simplicity the observer is placed opposite the center of the tight rope at a distance of one unit. It will be assumed that the observer can sense the angular deflection of his head with a standard deviation of 0.1 radians. The mathematical model for the observer is given below.

$$z(k) = \tan^{-1}(x(k)) + v(k) \quad (111)$$

$$E[v(k)] = 0 \quad (112)$$

$$E[v(k)v(j)] = \begin{cases} R & \text{for } j=k \\ 0 & \text{for } j \neq k \end{cases} \quad (113)$$

and R has been taken as 0.01.

Finally the a priori data must be specified. For this example the use of an artificial a priori will be illustrated. The physical situation is such that before taking any data there is essentially no information available about position of the tight rope walker. This fact is modeled by taking the a priori estimate as zero and assigning a very large variance to this estimate, say 10,000.

The structure is

$$x(1/0) = 0 \quad (114)$$

$$P(1/0) = 10,000 \quad (115)$$

This completes the mathematical model of the physical process underlying the observations. Assume that the first two observations are 0.7854 and 0.900 radians. Using these observations the computations required by the filter are described in detail.

The method described in Section 5 for the single-stage case is applied to this example for the first observation. The first estimate of the state is the a priori estimate $x_1^1 = 0$. The partial derivative of the measurement is evaluated to find $H^1 = 1/[1+(x_1^1)^2] = 1$ for the

first iteration. Applying (44) yields

$$G^1 = (1) \cdot (10,000) / [(1) \cdot (10,000) \cdot (1) + (.01)] = 1.$$

Next z^1 is computed from (40),

$$z^1 = .7854 - \tan^{-1}(0) + (1) \cdot (0) = 0.7854.$$

From (43)

$$x_1^2 \text{ (not an exponent)} = 0 + (1) \cdot (.7854 - (1) \cdot (0)) = 0.7854$$

In Table I the results of repeating this procedure three times are shown.

Also shown in the table is the cost associated with each estimate, including the cost for the a priori. From a table for the chi square variable the threshold values are found to be $C_L(0.05) = 3.84$ and $C_M(0.95) = 0.004$. Both of these are for a single degree of freedom. C_L has one degree of freedom since there are two constraints, the a priori and the observation; less one adjustable quantity, the single component of the state estimate. The C_M also has a single degree of freedom since there is only one element in the state vector. From Table I it can be seen that the cost associated with x_1^2 might be considered a minimum cost, but there has been a significant decrease in the cost so the process is repeated. Considering the last iteration it may be said that 1.0 is not a significantly better estimate of the true state than 0.9767. This terminates the first minimization process.

It is interesting to note that the device of taking a large a priori variance has led to the expected result that $x(1/1)$ converges rapidly to $\tan [z(1)] = 1.0$. It may occur to the reader that this is the hard way to evaluate $\tan [z(1)]$, but the advantage of general applicability of the method outweighs the advantages of considering special cases. In any case, the machinery for handling a priori information must be available in order to implement the lumping procedure.

TABLE I

EXAMPLE OF SINGLE-STAGE MINIMIZATION PROCESS

A PRIORI ESTIMATE = 0.0

OBSERVATION = .7854

A PRIORI VARIANCE = 10,000

OBSERVATION
ERROR = .01
VARIANCE

ITERATION NUMBER	OLD ESTI- MATE				NEW ESTI- MATE	COST
i	x_1^i	F^i	G^i	z^i	x_1^{i+1}	$C(x^i)$
1	0.000	1.000	1.000	0.7854	0.7854	61.68
2	0.7854	0.6185	1.617	0.6042	0.9767	1.40
3	0.9767	0.5118	1.954	0.5121	1.0000	0.015
4	1.0000					.00001

Next the bias and variance of the nonlinear noise are evaluated to determine whether a lumping operation is indicated. The nonlinear noise depends upon the variance of the new estimate and the second derivative of the observation function evaluated at the estimate. The variance of this estimate is computed from (54):

$$P_1 = 10,000 - (10,000)^2 / (40,000 + 0.01) = 0.04.$$

Using this variance the lumping test criterion can be computed from Appendix II. The bias due to the second order terms is 0.01 and the variance of the second order term is 0.0001. Comparing this with the variance of natural noise (observation errors) it is noted that there is an order of magnitude difference and a lumping operation would normally take place.

For this example the first stage will not be lumped so that the details of the two-stage minimization process may be illustrated. If lumping had taken place the estimate would be projected forward to form the a priori estimate for the next time frame. Since $\Phi = 1$ for this simple dynamic system the new a priori is just the old best estimate. From (21) the variance of the a priori estimate is .06. To obtain the estimate of the position of the tight rope walker at the second time frame one proceeds exactly as above using the new a priori information.

In order to solve the two-stage minimization problem the problem is reduced to a single-stage problem. The elements of the single-stage state are the position at the first time and the position at the second time. The relations described in Section 6 are used to generate a complete single-stage problem having a two-dimensional state.

$$x_0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \equiv \begin{bmatrix} x_{1_0} \\ x_{2_0} \end{bmatrix} \quad (116)$$

$$P_o = \begin{bmatrix} 10000 & 10000 \\ 10000 & 10000.02 \end{bmatrix} \quad (117)$$

$$z = \begin{bmatrix} 0.7854 \\ 0.9000 \end{bmatrix} \quad (118)$$

$$R = \begin{bmatrix} .01 & 0 \\ 0 & .01 \end{bmatrix} \quad (119)$$

$$h(x) = \begin{bmatrix} h_1(x) \\ h_2(x) \end{bmatrix} = \begin{bmatrix} \tan^{-1}(x_1) \\ \tan^{-1}(x_2) \end{bmatrix} \quad (120)$$

The above is a new single-stage minimization problem and conceptually, it is solved in exactly the same way that the previous (one dimensional) problem was solved, i.e., by repeated application of (43). As a practical matter it is expedient to adjust the estimate separately for each element in z . This is possible since the errors in the observations are independent. This is true in general because the errors were assumed to be white in the multi-stage problem.

Each iteration proceeds in two steps. First both elements of the estimate are adjusted for the first element of z and then the resulting adjusted estimates are adjusted for the second element of z . See Figure 3. The result of the first adjustment is already known and need not be computed. The first element of this intermediate result is the best estimate from the previous minimization process. The second element is just the predicted estimate $x(2/1) = \Phi x(1/1) = 1.0$ from (83). The intermediate covariance is computed from (85), (86), and (87) and known value of $P(1/1)$.

$$x^1_{\text{inter.}} = \begin{bmatrix} 1.0 \\ 1.0 \end{bmatrix} \quad (121)$$

$$P_{\text{inter.}} = \begin{bmatrix} 0.04 & 0.04 \\ 0.04 & 0.06 \end{bmatrix} \quad (122)$$

Starting from this intermediate result the adjustment to the estimate for the second measurement is computed. H^1 is now a row vector of partials of the second observation function with respect to both elements of the state vector: $H^1 = [0. \quad 0.50]$. Note that the zero in H^1 is a general result of the way in which the problem is formulated. The measurement function is formally a function of the whole state x although it is clear from the construction of the single-stage form that each individual element of the measurement function, $h(x)$, is a function of the state of the system at only one time. From (44) the adjustment coefficients, G^1 , are obtained.

$$G^1 = \begin{bmatrix} .8 \\ 1.2 \end{bmatrix} \quad (123)$$

Applying (40) the synthetic observation is found to be

$$z^1 = .9 - \tan^{-1}(1) + (.5) \cdot (1) = .6146$$

and the adjusted estimates are

$$x_1^2 = \begin{bmatrix} 1.0917 \\ 1.1375 \end{bmatrix}. \quad (124)$$

The cost for the intermediate estimate (121) and the above (124) estimate, x_1^2 , are 1.313 and .552 respectively. Since these are a sum of four squared residuals with two adjustable quantities the convergence-test parameters are different. $C_L(.05) = 5.99$ and $C_M(.05) = .103$. Based on these parameters it may be inferred that the above estimate is significantly better than the intermediate estimate. A second iteration will be carried out.

The minimization is accomplished in two steps. The first step is to adjust the a priori vector for the first observation. The difference between this step and the single stage minimization is that the partial derivatives are evaluated at x_1^2 given above in (124). The result of this step is comparable to the previous intermediate estimate in (121) and (122)

$$x_{\text{inter.}}^2 = \begin{bmatrix} .9951 \\ .9951 \end{bmatrix} \quad (125)$$

$$P_{\text{inter.}} = \begin{bmatrix} .048 & .048 \\ .048 & .068 \end{bmatrix} \quad (126)$$

The second step is to adjust this intermediate estimate for the second observation. The partial derivative, H , is to be evaluated at x_1^2 and not $x_{\text{inter.}}^2$. The result of this second adjustment is

$$x_1^3 = \begin{bmatrix} 1.0978 \\ 1.1405 \end{bmatrix} \quad (127)$$

The cost evaluated at this estimate is .539. The convergence tests indicate that the last estimate is not significantly better than the previous one. This minimization is said to have converged.

After it has been decided that the process has converged it is necessary to evaluate the second-order terms in the expansion of the nonlinear measurement function. For this purpose it is necessary to have the covariance of the last estimate. This covariance is automatically computed by the method displayed in Figure 3.

$$P_1 = \begin{bmatrix} .02096 & .02096 \\ .02096 & .02966 \end{bmatrix} \quad (128)$$

From Appendix II the expected value of the second order term, denoted as the bias or simply b , is

$$b = \begin{bmatrix} .0048 \\ .0065 \end{bmatrix}, \quad (129)$$

and the variance about this expected value is

$$R' = \begin{bmatrix} .000023 & - - \\ - - & .000042 \end{bmatrix}. \quad (130)$$

Assume that only the first of the measurements has negligible second-order terms. Then a lumping operation is indicated. This might be accomplished by noting the two-stage interpretation of x_{inter}^2 and P_{inter} .

$$x_{inter}^2 \equiv \begin{bmatrix} x(1/1) \\ x(2/1) \end{bmatrix} \quad (131)$$

and

$$P_{inter} \equiv \begin{bmatrix} P(1,1/1) & P(1,2/1) \\ P(2,1/1) & P(2,2/1) \end{bmatrix} \quad (132)$$

The lower element of x_{inter}^2 and the lower-right element of P_{inter} constitute the a priori information for the state at the second time frame.

It would be possible to consider this a priori information and the second observation as a new problem.

In the computer program it is inconvenient to store all of the intermediate results awaiting a decision on which stages are to be lumped. An alternate method for carrying out the lumping operation will be described. Assume as above that it has been decided to lump the first stage. The next steps in the filter operation would normally be as follows. The third state is predicted using (83) and the covariance matrix augmented accordingly using (84). These estimates are adjusted for the

third observation. The main minimization procedure is begun. Recall that the main minimization procedure is carried out in three steps, an adjustment for each observation. After the first of these steps the the intermediate result can be interpreted as

$$x_{\text{inter.}} = \begin{bmatrix} x(1/1) \\ x(2/1) \\ x(3/1) \end{bmatrix} \quad (133)$$

and

$$P_{\text{inter.}} = \begin{bmatrix} P(1,1/1) & P(1,2/1) & P(1,3/1) \\ P(2,1/1) & P(2,2/1) & P(2,3/1) \\ P(3,1/1) & P(3,2/1) & P(3,3/1) \end{bmatrix}. \quad (134)$$

At this point the lumping operation is carried out by reducing the dimension of the single stage to two (see Figure 2) and storing the lower part of x_{inter} (133) and the lower-right part of P_{inter} (134) in the area assigned to a priori information. The remaining two steps of the main minimization procedure are then carried out. If the process has not converged then the next minimization will only have two steps.

This completes the description of the operation of the filter for this very simple example.

11. Target Tracking.

It was decided to exercise the scheme on as realistic a problem as could be found. The target data (which was fed into the filter) was generated by a sophisticated simulator. The target motion is the result of maneuver commands generated by the user of the simulation scheme. The simulator then computes the motion of the target, and simulates the radar returns which that target would generate. The simulated radar is of the search type having as available outputs range, range rate and three direction cosines at a rate of one frame every two seconds. The simulator decides, taking into account the relative position of the target and radar, whether a return is received. If a return is received, the simulator outputs a noisy version of the true range, range rate and direction cosines. If no return is received the simulator sets a flag in the output data. At long ranges the chances of getting a return are relatively small but as the range decreases the radar gets returns more and more consistently.

Forming the Mathematical Model

It should be noted that this problem, as sketched above, does not fall directly into the model which has been assumed in the development of the technique. Among the parameters which have not been given in the description of the problem are Φ , Γ , R and even x (the state space). This is typical of the way in which a problem is first encountered. What follows will be a series of engineering approximations which yield the mathematical model. This model forms the basis for the filter design.

First consider the stochastic dynamic model. The dynamic model may be viewed as specifying two features of the problem. The first is a prediction function. One asks: how would one predict some future state of the system given perfect knowledge of the present state? This question in

fact helps to define the concept of state. The state of the system (for filtering purposes) is that collection of current attributes of the system which has a bearing on the future of the system. For the aircraft target the assumption of straight and level flight leads to an assignment of position and velocity as the states. The prediction function is based upon the assumption of constant velocity. Thus the components of the state vector are

$$x_1 = \text{north position (miles)}$$

$$x_2 = \text{north velocity (miles/sec)}$$

$$x_3 = \text{east position (miles)}$$

$$x_4 = \text{east velocity (miles/sec)}$$

$$x_5 = \text{down position (miles)}$$

$$x_6 = \text{down velocity (miles/sec)}$$

and the prediction function is linear in the states and of the form

$x(k+1)_{\text{Pred}} = \Phi x(k)$. Φ is the discrete time form of three independent double integrators for a sample time of 2 seconds.

$$\Phi = \begin{bmatrix} 1 & 2 & \vdots & 0 & 0 & \vdots & 0 & 0 \\ 0 & 1 & \vdots & 0 & 0 & \vdots & 0 & 0 \\ \hline 0 & 0 & \vdots & 1 & 2 & \vdots & 0 & 0 \\ 0 & 0 & \vdots & 0 & 1 & \vdots & 0 & 0 \\ \hline 0 & 0 & \vdots & 0 & 0 & \vdots & 1 & 2 \\ 0 & 0 & \vdots & 0 & 0 & \vdots & 0 & 1 \end{bmatrix}$$

The second feature which the model must provide is a measure of the prediction errors, or equivalently Γ . This implies that the prediction errors are random variables made up of several normalized random variables.

The prediction error covariance is then $Q = \Gamma \Gamma^T$. For this problem Γ was chosen under the assumption that in each direction there would be a step-wise constant component of acceleration of random amplitude having

zero mean and variance σ^2 . This implied that Γ takes the form

$$\Gamma = 2 \begin{bmatrix} \sigma_N & 0 & 0 \\ \sigma_N & 0 & 0 \\ 0 & \sigma_E & 0 \\ 0 & \sigma_E & 0 \\ 0 & 0 & \sigma_D \\ 0 & 0 & \sigma_D \end{bmatrix} .$$

The parameters σ_N , σ_E and σ_D must be chosen to account for turbulence and pilot maneuvers, which, from the filter's point of view, appear as random accelerations.

$$\sigma_N = .02$$

$$\sigma_E = .02$$

$$\sigma_D = .0032$$

Secondly consider the observation process. Having chosen the state space x it is straightforward to write the functions $h(x)$

$$h_1(x) = (x_1^2 + x_3^2 + x_5^2)^{\frac{1}{2}} = \text{range (miles)}$$

$$h_2(x) = \frac{(x_1x_2 + x_3x_4 + x_5x_6)}{(x_1^2 + x_3^2 + x_5^2)^{\frac{1}{2}}} = \text{range rate (miles/sec)}$$

$$h_3(x) = \frac{x_1}{(x_1^2 + x_3^2 + x_5^2)^{\frac{1}{2}}} = \text{north direction cosine}$$

$$h_4(x) = \frac{x_3}{(x_1^2 + x_3^2 + x_5^2)^{\frac{1}{2}}} = \text{east direction cosine}$$

$$h_5(x) = \frac{x_5}{(x_1^2 + x_3^2 + x_5^2)^{\frac{1}{2}}} = \text{down direction cosine}$$

The noise variance was approximated from considerations related to the radar simulator.

$$R = \begin{bmatrix} 1.5 & 0 & 0 & 0 & 0 \\ 0 & 4.0 \times 10^{-6} & 0 & 0 & 0 \\ 0 & 0 & 1.8 \times 10^{-5} & 0 & 0 \\ 0 & 0 & 0 & 3.1 \times 10^{-5} & 0 \\ 0 & 0 & 0 & 0 & 1.5 \times 10^{-4} \end{bmatrix}$$

Finally, the a priori estimate and its associated covariance must be specified in order to complete the mathematical model. The fact that the first radar return has been received places the target in a certain volume in physical space. The position components of the a priori estimate were taken as the centroid of that volume and the limits of the volume were taken as three standard deviations on either side of the centroid. The a priori velocity estimates were based on the assumption that the target was headed directly toward the radar (in the negative north direction). The magnitude of the velocity was taken as that of a Mach 2 target. The covariance of each velocity estimate was assumed to be large compared to the square of this velocity. The a priori estimate was taken as:

$$x(1/0) = \begin{bmatrix} 80.0 \\ -0.3 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} .$$

The a priori covariance was taken as:

$$P(1/0) = \begin{bmatrix} 400 & 0 & 0 & 0 & 0 & 0 \\ 0 & .09 & 0 & 0 & 0 & 0 \\ 0 & 0 & 400 & 0 & 0 & 0 \\ 0 & 0 & 0 & .09 & 0 & 0 \\ 0 & 0 & 0 & 0 & 400 & 0 \\ 0 & 0 & 0 & 0 & 0 & .09 \end{bmatrix}$$

This completes the process of abstracting the physical situation into the form of the mathematical model.

It should be emphasized that the abstraction process must be carried out for each physical system that generates a sequence of measurements. If the model accurately describes the conditions under which the measurements are made then the filter can be expected to yield estimates which are best in some sense. Even an accurate model and an optimum filter do not assure that the estimates will be adequate for any particular purpose.

The Algorithm Parameters

There are three parameters that define the iteration termination criteria. They are the probability, α , that a minimum cost is greater than a given threshold, C_L ; the level of statistical significance, β ; and the number of significant digits used by the computer.

The threshold, C_L , depends upon the number of stages carried (which determines the number of degrees of freedom) as well as upon α . There are five degrees of freedom in the cost for each stage carried since the system dynamics (1) introduce six constraints and the observations (5) introduce five constraints and there are but six adjustable parameters (the components of the state vector) for each stage carried. An α of 0.05 was chosen in order to have a small but finite number of cases where the minimum cost was greater than C_L .

The likelihood-ratio-test threshold, C_M , also depends upon the number of stages carried. The number of degrees of freedom is six for each stage carried since that is the number of components in the state vector. A significance level of 0.95 was arbitrarily chosen. The thresholds C_L and C_M are shown in Table II.

The filter was implemented on a CDC 1604 computer. This machine carries about ten significant figures. Two successive iterations were considered to be equivalent if all of the components of the estimate were equal in the first nine significant figures.

The lumping criterion is based on a comparison of the covariance of the observation errors and the covariance of the nonlinear noise introduced by the linearization process. For concreteness, the nonlinear noise was considered to be negligible when its covariance was less than that of the observation errors by a factor of ten.

Target Tracking Results

Three target trajectories were filtered. The results were quite similar. The target on which the largest number of observations were received will be described in some detail.

Figures 5 through 9 are a graphical display of the filter operation. Figures 5 and 6 show the true target trajectory projected on the NORTH-EAST plane and the NORTH-DOWN plane. Superimposed on the true trajectory are confidence areas generated by the filter. The boxes are used to provide a measure of the quality of the estimate. The size and shape of the box is computed from the covariance matrix of the estimate. If the estimation errors were Gaussian with a covariance equal to that computed by the filter the box would have the following interpretation. There is an ellipse, centered about the estimate which contains the true state

TABLE II
 ITERATION CONTROL PARAMETERS AS A FUNCTION OF THE NUMBER
 OF STAGES CARRIED FOR $\alpha = .05$ AND $\beta = .95$

NUMBER OF STAGES CARRIED	C_L	C_M
1	11.07	1.64
2	18.31	5.23
3	25.00	9.39
4	31.41	13.85
5	37.65	18.49
6	43.77	23.02
7	49.55	27.86
8	55.76	32.85
9	61.33	37.80

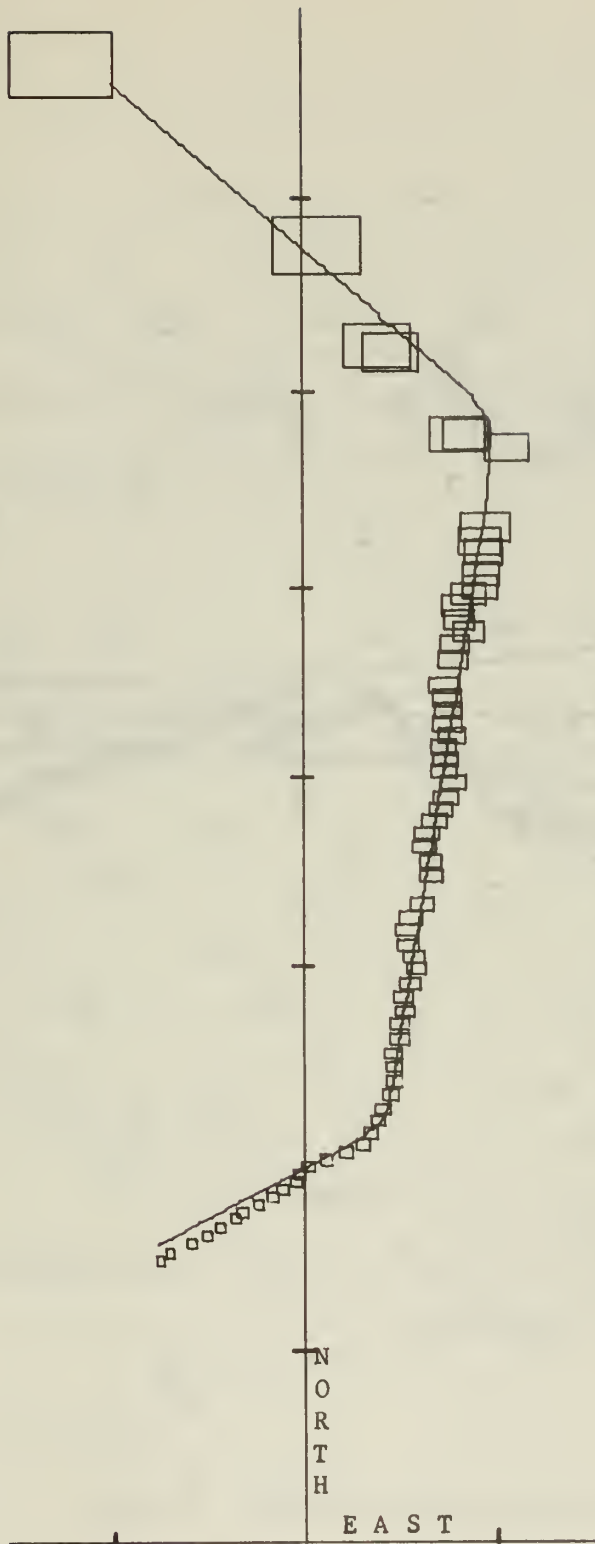


Fig. 5. Projection on NORTH-EAST
plane of true trajectory
with estimates.

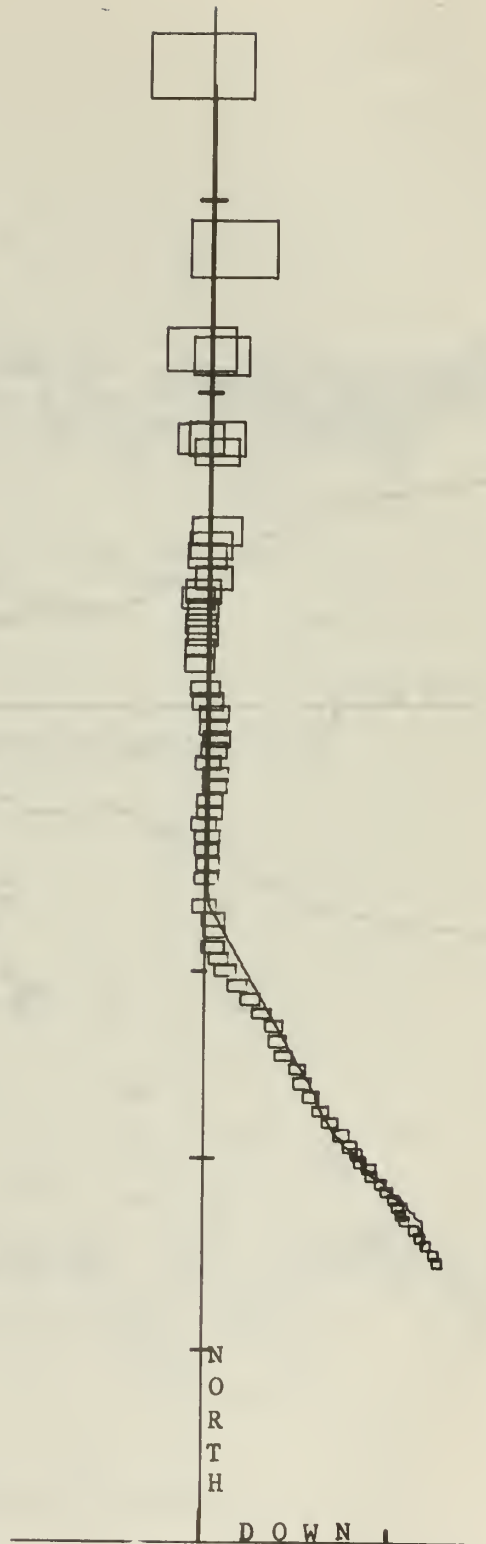


Fig. 6. Projection on NORTH-DOWN
plane of true trajectory
with estimates.

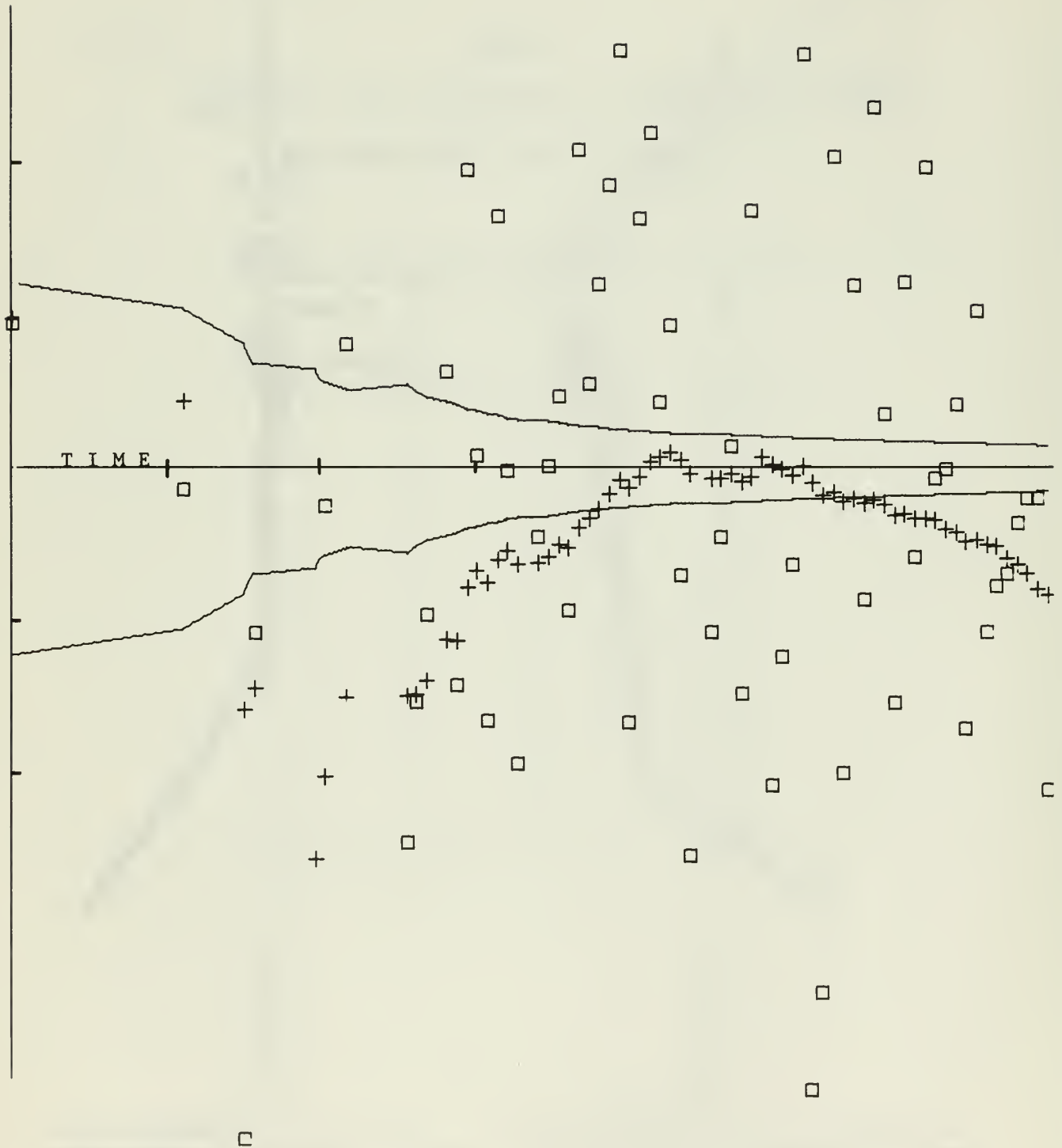


Fig. 7. Observation (\square) and estimation ($+$) errors in NORTH coordinate.

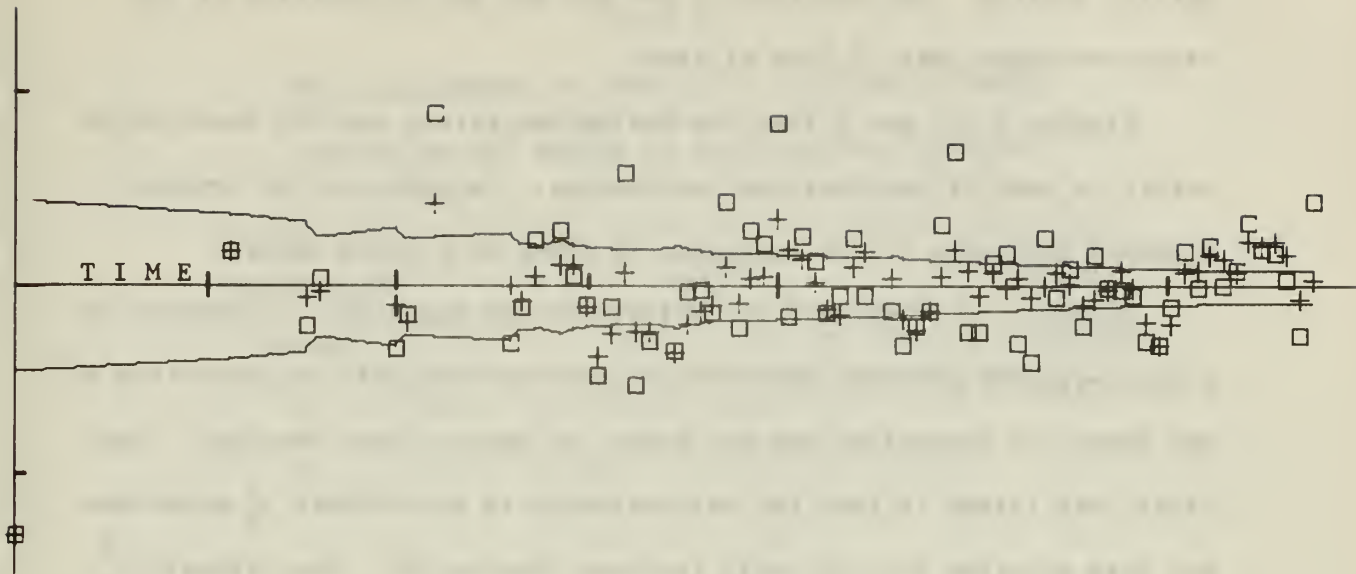


Fig. 8. Observation (□) and estimation (+) errors in EAST coordinate.

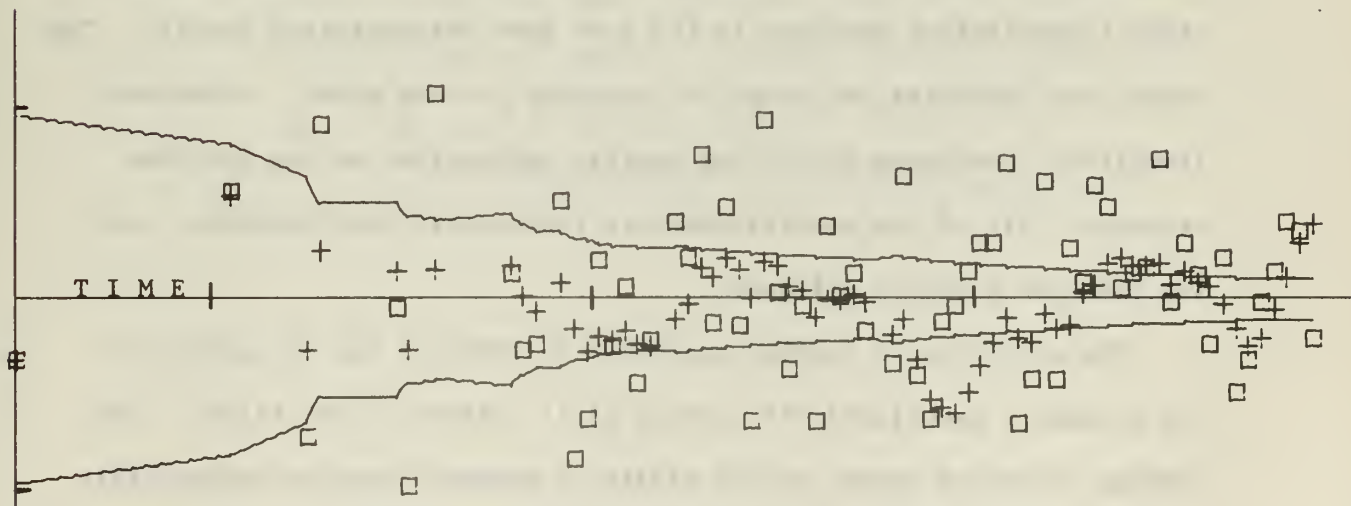


Fig. 9. Observation (□) and estimation (+) errors in DOWN coordinate.

with probability of 0.63 and whose boundary is a curve of constant probability density. The vertices of the box are the extremities of the major and minor axes of that ellipse.

Figures 7, 8, and 9 show the estimation errors and the observation errors in each of the position coordinates. In addition the computed standard deviation of the estimates is shown as a solid curve.

In order to illustrate the ability of the algorithm to converge to a least-squares estimate the cost is given in Table III as a function of the number of iterations and the number of observations received. The first cost listed in each row was evaluated at an estimate x_1^1 which has not been adjusted for the newly received observation. The estimate x_1^1 for a (k+1)-stage minimization process is given by (89). The first iteration yields x_1^2 by adjusting x_1^1 for the most recently received observation. Only the partial derivatives associated with this new observation are evaluated for this step. This step is comparable to the simple single-stage linearization employed in [4] with such disappointing results. The second row indicates the danger of stopping at this point. Subsequent iterations reevaluate all of the partial derivatives at the previous estimate. All of the observations are reprocessed with estimates starting from the a priori estimate.

The cost after a lumping operation is only the sum of squares of the residuals associated with stages still carried by the filter. The lumping operation occurs in the middle of second iteration because this is the first time that the intermediate results needed to form the new a priori estimate of the remaining stage become available again after the decision to lump has been made.

The time required to produce a least-squares estimate is tabulated in Table IV. The time indicated does not include the time required for

TABLE III

TYPICAL SEQUENCE OF COSTS AS A FUNCTION OF NUMBER
ITERATIONS AND NUMBER OF OBSERVATIONS PROCESSED

OBSERVATION NUMBER	$C(x_1^i)$				STAGES
	i=1	i=2	i=3	i=4	CARRIED
1	367.	0.57	0.13		1
2	377.8	118.5	2.07	2.07	2
3	35.4	11.7	11.16		3
4	19.0	12.9	12.9		4
5	48.7	20.99	20.99		5
6	121.6	27.36	27.36		6
7	811.9	38.28	38.06		7
8	749.9	101.3	44.16	44.16	8
9	48.51	44.4	9.49*	9.49	2
10	12.27	10.98	1.51*	1.51	1

* A lumping operation occurred.

TABLE IV
 TYPICAL COMPUTATIONAL TIME REQUIREMENTS FOR
 THE CDC 1604 COMPUTER

OBSER- VATION NUMBER	COMPU- TATION TIME (SEC)	CUMULATIVE COMPUTATION TIME (SEC)	OBSERVATION ARRIVAL TIME (SIMULATED)
1	0.817	0.817	0.0
2	2.633	36.633	34.00
3	2.617	48.617	46.00
4	3.683	52.300	48.00
5	5.633	65.633	60.00
6	7.633	73.266	62.00
7	10.45	83.716	66.00
8	26.083	109.800	78.00
9	14.900	114.700	80.00
10	2.100	116.800	82.00
11	1.367	118.167	86.00
22	1.467	134.050	110.00
34	1.300	148.417	130.00
63	1.483	196.017	196.00

auxiliary computations performed for diagnostic purposes nor the time required to read the data from the magnetic tape. It does include all computations inherent in the filter operation such as evaluating the cost and covariance of the nonlinear noise. The cumulative running time has been adjusted to reflect the fact that the filter cannot begin to compute a new estimate until the next observation is available.

Comparison between Table III and Table IV yields the obvious fact that the time required to compute the estimate is highly dependent on the number of stages carried. From an analysis of the computations involved in Figure 3 it can be shown that the computations increase as the square of the number of stages times the system order. The computation time depends linearly on the number of observations.

The operation of the filter indicated for the tenth observation is typical of all of the remaining stages in both number of iterations and processing time required with the exception of the cases where the minimum cost was greater than C_L . This happened 3 times out of 67 observations on the longest run. In each case additional iterations involved only a single-stage. Two or three extra iterations were needed to satisfy the termination criterion. The largest relative change in any component of the estimate decreased approximately two orders of magnitude after each iteration. The average processing time for these three cases was about 1.9 seconds.

12. Conclusions

An algorithm has been developed for the processing of a sequence of noisy, nonlinear observations made on a dynamic system whose state is a random function of time. The best estimate of the state of the system at each observation time is defined to be the weighted-least-squares estimate.

This estimate is computed by solving a sequence of linear problems which approximate the nonlinear problem more and more closely. A method has been developed for automatically determining the number of iterations required to compute the least-squares estimate by the above procedure.

The computation of each estimate is based on a least-squares fit on only a finite sequence of past observations. A method has been developed for determining the length of this sequence of past observations. The information contained in the older observations is carried forward in the form of an a priori estimate.

The radar tracking problem is an example of the type of problem which falls within the scope of this investigation. The algorithm was implemented on a digital computer and used to process a sequence of observations provided by a realistic radar-target simulator. The estimation errors were generally within the expected range, considering the randomness of the dynamic system and the observation errors. The algorithm achieved the least-squares estimate in three or four iterations. The length of the sequence of observations on which the least-squares fit was based, rapidly settled to only a single previous observation. The computational requirements appear excessive when compared with those associated with linear observations. There are, however, no other generally applicable methods when the observations are nonlinear and there is little prior information about the state of the system.

BIBLIOGRAPHY

1. Bodewig, E. Matrix Calculus, 2nd ed. Interscience Publishers Inc., New York, 1959.
2. Cox, H. Recursive nonlinear filtering. Proceedings of the National Electronics Conference. v. XXI, 1965: 770-775.
3. Cox, H. On estimation of state variables and parameters. Presented on the Joint AIAA-IMS-SIAM-ONR Symposium on Control and System Optimization, Naval Postgraduate School, Monterey, 27-29, Jan., 1964.
4. Demetry, J. S. and R. E. Hudson. Application of the Kalman filter to spherical discrete track smoothing and prediction. Naval Postgraduate School. Research Paper no. 70, Sept., 1966.
5. Denham, W. F. and S. Pines. Sequential estimation when measurement function nonlinearity is comparable to measurement error. Analytical Mechanics Associates, Ind. Report prepared for AIAA 3rd Aerospace Sciences Meeting, New York, 24-26 Jan., 1966.
6. Deutsch, R. Estimation Theory. Prentice Hall, Englewood Cliffs, 1965.
7. Kalman, R. E. A new approach to linear filtering and prediction problems. Journal of Basic Engineering, ASME Transactions. 82-D, 1960: 35-45.
8. Kalman, R. and R. Bucy. New results in linear filtering and prediction theory. Journal of Basic Engineering, Transactions ASME. Series D, v. 83, no. 1, March, 1961: 95-108.
9. Lee, R. C. K. Optimal Estimation, Identification, and Control. M.I.T. Press, Cambridge, 1964.
10. Minka, K. Orbit determination and analysis by the minimum variance method. Martin Company Baltimore Division. Prepared for Air Force Cambridge Research Laboratories, Aug. 1965.
11. Ohap, R. F. and A. R. Stubberud. A technique for estimating the state of a nonlinear system. IEEE Transactions on Automatic Control. v. AC-10, April, 1965: 150-155.
12. Peschon, J. Review and extension of real-time estimation techniques. Stanford Research Inst. Paper, 1965.
13. Rauch, H. E. Solutions to the linear smoothing problem. IEEE Transactions on Automatic Control. v. AC-8, Oct., 1963: 371-373.

Appendix I

Discussion of singularity of P_0 .

The covariance matrix P_0 is assumed to be non-singular throughout the development, however, there is a meaningful interpretation for the case where P_0 is singular. It will be shown that the filtering equations are still valid in view of this interpretation and that in the one instance where the inverse of P_0 is required (in evaluating C) the use of the pseudo inverse is appropriate.

This development begins by defining a new set of state variables, y , so that the errors in the a priori estimates are uncorrelated.

$$y = Ux \quad (1)$$

$$y_0 = Ux_0 \quad (2)$$

where U is a unitary matrix such that $U^{-1} = U^T$ and

$$UP_0U^T = D \quad (3)$$

where D is a diagonal matrix.

It will be shown now that D is the covariance of the a priori estimates in the y states.

$$\begin{aligned} \text{Cov}(y_0) &\equiv E[(y-y_0)(y-y_0)^T] \\ \text{Cov}(y_0) &= E[U(x-x_0)(x-x_0)^T U^T] \\ \text{Cov}(y_0) &= UE[(x-x_0)(x-x_0)^T]U^T \\ \text{Cov}(y_0) &= D \end{aligned} \quad (4)$$

If P_0 is singular then D has at least one zero on the diagonal or, to be more specific, the rank of P_0 is equal to the rank of D which is the number of non-zero elements along the diagonal of D . There is no loss in generality in assuming that all of the non-zero elements of D are in the upper part of the diagonal. The upper elements of y_0 are

conventional statistical estimates of the corresponding elements of the true state y , having a variance given by the element in D . On the other hand, the lower elements of y_0 are precise or exact estimates of the corresponding true state components and have no variance. When these interpretations are reflected back to the x states the meaning of a singular P_0 becomes clear. A singular P_0 implies that a certain number of linear combinations of the states are known exactly.

It will now be shown that the filtering equations, used repeatedly in the minimization process, produce estimates consistent with these interpretations. That is, the estimate of those linear combinations of the states which were known exactly before adjustment for observed data are not affected by the adjustment process. Further the covariance matrix, P_1 of the adjusted estimates reflects the fact that these linear combinations are known exactly. This demonstration will be carried out using the y state space. The filter equations are transformed to operate on the y coordinates.

Consider first

$$x_1 = x_0 + P_0 H^T [HP_0 H^T + R]^{-1} [z - Hx_0] , \quad (5)$$

substituting y for x in (5) yields

$$U^T y_1 = U^T y_0 + P_0 H^T [HP_0 H^T + R]^{-1} [z - HU^T y_0] , \quad (6)$$

pre-multiplying both sides of (6) yields

$$y_1 = y_0 + UP_0 H^T [HP_0 H^T + R]^{-1} [z - HU^T y_0] , \quad (7)$$

substituting $U^T D U$ for P_0 from (3)

$$y_1 = y_0 + DUH^T [HU^T DUH^T + R]^{-1} [z - HU^T y_0] , \quad (8)$$

and finally making a change of variables $K \equiv HU^T$ yields the filtering equation,

$$y_1 = y_0 + DK^T [DKK^T + R]^{-1} [z - Ky_0] , \quad (9)$$

in the y states. Since the lower elements of D are zero it is clear that there is no change in these components of the adjusted estimate, y_1 , after application of (9).

Now consider the covariance equation

$$\begin{aligned} P_1 &= P_0 - P_0 H^T [H P_0 H^T + R]^{-1} H P_0 \\ P_1 &= U^T D U - U^T D U H [H U^T D U H^T + R]^{-1} H U^T D U \\ P_1 &= U^T [D - D K^T [K D K^T + R]^{-1} K D] U \end{aligned} \quad (10)$$

multiplying in front by U and in back by U^T

$$U P_1 U^T = D - D K^T [K D K + R]^{-1} K D \quad (11)$$

Thus P_1 reflects the fact that those linear combinations of x which were known exactly are still known exactly.

In the expression for the cost consider only the first term,

$\|x_0 - x_1\|_{W_1}^2$, where W_1 was assumed to be the inverse of P_0 if it existed.

Substituting in the y states yields

$$\begin{aligned} \|x_0 - x_1\|_{W_1}^2 &= \|U^T (\psi_0 - \psi_1)\|_{W_1}^2 \\ \|x_0 - x_1\|_{W_1}^2 &= \|\psi_0 - \psi_1\|_{U W_1 U^T}^2 \end{aligned}$$

Defining $Q \equiv U W_1 U^T$ and substituting above yields

$$\|x_0 - x_1\|_{W_1}^2 = \|\psi_0 - \psi_1\|_Q^2$$

Define $\tilde{y} \equiv (\psi_1 - \psi_0)$ and partition \tilde{y} into two subvectors

$$\tilde{y} \equiv \begin{bmatrix} \tilde{y}_u \\ \tilde{y}_l \end{bmatrix} = \begin{bmatrix} \tilde{y}_u \\ 0 \end{bmatrix}$$

where the lower subvector $\tilde{y}_l \equiv 0$ from (9)

Let Q be partitioned as

$$Q = \begin{bmatrix} D_u^{-1} & - \\ - & - \end{bmatrix}$$

where D_u is the upper, non-zero, part of D . Thus sum of the residuals can be expressed as

$$\|x_0 - x_1\|_{W_1}^2 = \|\tilde{v}_u\|_{D_u^{-1}}^2$$

The blank submatrices in Q above are immaterial since they will be multiplied by $\tilde{v}_\ell = 0$. Arbitrarily assigning zero submatrices to the blanks implies that those residuals which are known to be zero are given zero weight. This leads to

$$Q = \begin{bmatrix} D_u^{-1} & 0 \\ 0 & 0 \end{bmatrix} = U W_1 U^T$$

and premultiplying by U^T and postmultiplying by U yields

$$W_1 = U^T \begin{bmatrix} D_u^{-1} & 0 \\ 0 & 0 \end{bmatrix} U$$

but this is just an expression for the pseudo inverse of P_0 .

Thus if the pseudo inverse of P_0 is used in the definition of the cost there will be no weighting of the residuals in certain linear combinations of the states. On the other hand, if the x_1 is always computed using (5) or one of its derivatives then these particular residuals will always be zero.

During the discussion of the minimization algorithm the non-singularity P_0 was an important assumption. The discussion is valid for

the case of P_0 singular in the sense that a new minimization problem can be defined in terms of y_u , taking $y_{l_0} \equiv y_{l_1}$. The discussion then implies that the change in the estimate of y_u has a positive component in the direction of the negative gradient of C with respect y_u . When these conclusions are reflected back to the x state space it can be seen that the change in the estimate is related to the projection of the gradient of C into that subspace of the state space about which there is some uncertainty and for which an adjustment in the estimate is meaningful.

Appendix II

Consider a random variable

$$v = x^T A x \quad (1)$$

where A is a symmetric matrix and x is a Gaussian random variable with mean zero and covariance P . It is desired to find an expression for the first and second moment of the random variable v . The development begins by making a change of variables

$$x = B U y \quad (2)$$

where B is a decomposition of P such that $P = B B^T$, U is a unitary matrix as yet unspecified, such that $U U^T = I$ and y is a random vector of zero mean and identity covariance. The random variable v is expressed in terms of y by substituting (2) in (1).

$$v = y^T U^T B^T A B U y \quad (3)$$

Now let U be chosen so that

$$U^T B^T A B U = D \quad (4)$$

where D is a diagonal matrix. Substituting (4) in (3) yields

$$v = y^T D y \quad (5)$$

or v can be expressed in terms of the components of y

$$v = \sum_i d_i y_i^2 \quad (6)$$

To evaluate the first moment of v the order of summation and expectation is interchanged.

$$E[v] = E\left[\sum_i d_i y_i^2\right]$$

$$E[v] = \sum_i d_i E[y_i^2]$$

The components of y all have unit variance.

$$E[v] = \sum_i d_i \quad (7)$$

Expressing (7) in matrix form yields

$$E[v] = \text{tr}D \quad (7')$$

The second moment is evaluated by expanding v^2 in terms of the components of y .

$$\begin{aligned} E[v^2] &= E\left[\left(\sum_i d_i y_i^2\right)\left(\sum_j d_j y_j^2\right)\right] \\ E[v^2] &= E\left[\sum_i d_i^2 y_i^4 + \sum_{i \neq j} d_i d_j y_i^2 y_j^2\right] \\ E[v^2] &= \sum_i d_i^2 E[y_i^4] + \sum_{i \neq j} d_i d_j E[y_i^2 y_j^2] \end{aligned} \quad (8)$$

The first term is evaluated by recalling that the fourth central moment of a unit-variance Gaussian random variable is 3. Each element of the second term can be factored due to the independence of the components of y .

$$\begin{aligned} E[v^2] &= 3 \sum_i d_i^2 + \sum_{i \neq j} d_i d_j E[y_i^2] E[y_j^2] \\ E[v^2] &= 3 \sum_i d_i^2 + \sum_{i \neq j} d_i d_j \\ E[v^2] &= 2 \sum_i d_i^2 + \left(\sum_i d_i\right)^2 \end{aligned}$$

Substituting (7) above yields

$$E[v^2] = 2 \sum_i d_i^2 + (E[v])^2 \quad (9)$$

This implies that the variance of v is

$$\text{Var}[v] = 2 \sum_i d_i^2 \quad (10)$$

or in matrix form

$$\text{Var}[v] = 2\text{tr}[D^2]. \quad (10')$$

The results, (7) and (10), are expressed in terms of the parameters of the original problem. Substituting (4) in (7') yields

$$E[v] = \text{tr}[U^T B^T A B U].$$

Taking advantage of the fact that $\text{tr}[AB] = \text{tr}[BA]$ yields

$$E[v] = \text{tr}[A B U U^T B^T]$$

and cancelling the unitary matrices yields

$$E[v] = \text{tr}[AP]. \quad (11)$$

The development for the variance proceeds along analogous lines.

$$\text{Var}[v] = 2\text{tr}[U^T B^T A B U U^T B^T A B U]$$

$$\text{Var}[v] = \text{tr}[A B B^T A B B^T]$$

$$\text{Var}[v] = \text{tr}[APAP] \quad (12)$$

It is possible to consider the covariance of two random variables as follows

$$v \equiv x^T A x \quad (1)$$

and a second random variable

$$v' \equiv x^T A' x \quad (1')$$

The means of both of these random variables are known from (11) and an expression will be developed for the expected value of the product. After making the same change of variables as before, the expected value of the product can be expressed as

$$E[v'v] = E[y^T C y y^T D y] \quad (13)$$

where

$$C = U^T B^T A' B U \quad (14)$$

This is equivalent to

$$E[v'v] = \text{tr}(C E[y y^T D y y^T]) \quad (15)$$

Examining only the elements of the matrix inside the expected value operator, it is noted that the middle term, $y^T D y$, is a scalar.

$$y^T D y = \sum_i d_i y_i^2 \quad (16)$$

Thus, the elements of the matrix are

$$E[y_i y_j \sum_k d_k y_k^2] \quad (17)$$

This expected value is zero for $i \neq j$ and for $i=j$ it can be written as

$$\begin{aligned} E[y_i^2 \sum_k d_k y_k^2] &= E[d_i y_i^4] + E[y_i^2 \sum_{k \neq i} d_k y_k^2] \\ E[y_i^2 \sum_k d_k y_k^2] &= 3d_i + \sum_{i \neq k} d_k \\ E[y_i^2 \sum_k d_k y_k^2] &= 2d_i + \sum_k d_k \end{aligned} \quad (18)$$

Thus, the expected value is a diagonal matrix of the following form

$$E[y y^T D y y^T] = 2D + (\sum_k d_k) \cdot I$$

Substituting in (15) yields

$$E[v v] = \text{tr} C [2D + (\sum_k d_k) I]$$

which can further be simplified to

$$E[v v] = 2 \text{tr} [CD] + [\sum_k d_k] \text{tr} C \quad (19)$$

Considering only the factors of the last term, it is noted that

$$\sum_k d_k = E[v] \quad (21)$$

and that

$$\begin{aligned} \text{tr} C &= \text{tr}[U^T B^T A B U] \\ \text{tr} C &= \text{tr}[A P] \\ \text{tr} C &= E[v] \end{aligned} \quad (22)$$

Now the first term can be identified with the covariance from the general expression

$$\text{Cov} [x_1, x_2] = E[x_1 x_2] - E[x_1] E[x_2] \quad (23)$$

$$\text{Cov}[v'v] = 2\text{tr}[C D]$$

$$\text{Cov}[v'v] = 2\text{tr}[U^T B^T A' B U D]$$

$$\text{Cov}[v'v] = 2\text{tr}[A' B U D U^T B^T]$$

$$\text{Cov} [v'v] = 2\text{tr}[A' B U U^T B^T A B U U^T B^T]$$

$$\text{Cov} [v'v] = 2\text{tr}[A' P A P] \quad (24)$$

The useful results of this analysis are (11) and (24) since (12) is only a special case of (24). Since these results will be used as a guide even when the random variable x is not known to be Gaussian it is worth reviewing just where the assumptions were used. The factor 2 which appears in (24) comes directly from Gaussian assumption (i.e., the fourth central moment is 3 times the second central moment squared). A second result of the Gaussian assumption is that the new random variables y are statistically independent (used in (8) and (18)). The variables y are uncorrelated (since D is diagonal) but only in the Gaussian case does this imply independence. The effect of this assumption is difficult to evaluate. On the other hand (8) does not depend upon the Gaussian assumption.

Appendix III

..

Program Listings

PROGRAM RUN

```

COMMON PHI(8,8),DEL(8,8),Q(8,8),X10(8,10),P10(80,80),R(8),S10(8,8)
1,Y10(8),X(8,10),SQ(8,8),SR(8),Z(8,10),H(8,8,10),DELS(8,8),Z1(8,10)
2,H1(8,8,10),ZA(8,10),X1(8,10),X2(8,10),PB(80,80),NS,NW,NM,MAX,NU,
3 K,LSHOOT,LU,KMAX,COST1,COST2
4 ,RR(8,10)
5,ABC(8,8,10),BCD(8,8,10),NR(10),NSK(10)
PRINT 300
300 FORMAT (1H1)
REWIND 49
CALL READ
C INPUT DESIGN MATRICES THROUGH SUBROUTINE READ
C SKIP A PAGE
PRINT 300
C INITIALIZE VARIABLES
CALL INIT
NFIL=0
SSGAV=0.
K=1 $DO 4 I=1,NS $ DO 5 J=1,NS
5 PB(I,J)=P10(I,J)
4 X2(I,1)=X10(I,1)
COST1=0,$LU=0 $ CALL SHIFT
X(1,1)=0.
REWIND 3
READ TAPE 3, MAX
MAX=MAX*5
PRINT 116,MAX
JGA=0
WRITE TAPE 49,NS,NM,MAX
PRINT 300
X(1,2)=0.
C X(1,2) IS USED TO STORE THE TOTAL(CUMULATIVE ) COST OVER ALL STAGES
C MAIN FILTER LOOP STARTS HERE

```

```

DO 30 KM=1,MAX
DO 33 I=1,NS
33 RR(I,K)=R(I)
READ TAPE 3,ITMAX,JT
READ TAPE 3, GX      ,(A,I=2,ITMAX) ,GY      ,(A,I=2,ITMAX),GZ
READ TAPE 3,IB
READ TAPE 3, ((A,I=1,JT),J=1,IB), Z(3,K),(A,I=2,IB)
1      , Z(4,K),(A,I=2,IB)
1      , Z(5,K),(A,I=2,IB)
1      , Z(1,K),(A,I=2,IB)
1      , Z(2,K),(A,I=2,IB)
C CONVERT RANGE RATE TO MILES PER SEC
Z(2,K)=Z(2,K)/6076.103
116 FORMAT( 30H NUMBER OF DATA SETS PROCESSED, I3)

C TEST FRAME FOR VALID DATA
GA = ABSF(Z(2,K)) -10.
IF (GA)20,20,21
21 CONTINUE
IF(JGA)49,30,49
49 CONTINUE
C NO DATA THIS FRAME
NSK(K)=NSK(K)+1
CALL AUGMENT (P10,X10)
CALL AUGMENT (PB ,X2 )
DO 51 I=1,NS
IC=(K-1)*NS+I
DO 52 J=1,NS
IB=(K-1)*NS+J
DO 53 L=2,K
IA=(L-2)*NS+I
P10(IA,IB)=P10(IB,IA) =P10(IA,IB+NS)
53 PB (IA,IB)=PB (IB,IA)= PB(IA,IB+NS )
P10(IC,IB) =P10(IC+NS,IB+NS)
52 PB(IC,IB)= PB(IC+NS,IB+NS)
X10(I,K)=X10(I,K+1)

```



```

51 X2 (I,K)=X2(I,K+1)
   GO TO 30
20 CONTINUE
   C   IGNORE FRAMES WITH NEGATIVE RANGE
   IF (Z(1,K))21,21,24

24 CONTINUE
   IF (ISWF(2)) 500,501,500
   C   ALLOWS OPERATOR TO CONTROL FROM THE CONSOLE THE
   C   NUMBER OF FRAMES PROCESSED
501 CONTINUE
   JGA=1
   CALL TIME
   CALL SKIP DATA
   CALL RUN1
   CALL RUN2
   CALL TIME
   NSK(K+1) =1

   C   THIS IS USED AS AN INTUITIVE MEASURE OF FILTER QUALITY
   C   SMOOTHING IS BY A LOW PASS FILTER WITH TIME CONSTANT OF TEN FRAMES
   C   GENERATES SMOOTHED SUM OF SQUARES OF POSITION VARIABLES
   NFIL=NFIL+1
   IF (NFIL-10)26,27,27
26 A=NFIL $ GAIN=1./A
27 SSQAV=SSQAV+GAIN*
   1 ((X2(1,K)-GX)**2+(X2(3,K)-GY)**2+(X2(5,K)-GZ)**2-S
   1 SQAV)
   AG= NFIL-K
   AG=AG* 5.-X(1,2)
   AG =(CUBERTF(X(1,1)/AG)-1.+ 2./((AG*9.))/SQRTF(2./(9.*AG)))
   C   COMPUTE THE NORMAL APPROXIMATION TO THE CHI SQUARE VARIABLE
   C   DISTRIBUTION, THIS IS USED TO EVALUATE THE OVERALL CONSISTANCY
   C   OF THE RECIEVED DATA WITH THE MODEL ASSUMED

22 CONTINUE

```

```

C   OUTPUT INFORMATION ON FILTERED ESTIMATES FOR GRAPHICAL DISPLAY
C   ON FRAMES WITH DATA ONLY
C   STORE INFORMATION FOR FILTER SUMMARY
      WRITE TAPE 49, NFIL, KM, K, SSQAV, AG
      WRITE TAPE 49, (Z(I,K), I=1,5)
      WRITE TAPE 49, (X2(I,K), I=1,6)
      LN=K*NS
      LI=LN-NS+1
      WRITE TAPE 49, ((PB(L,I), L=LI, LN), I=LI, LN)
      WRITE TAPE 49, GX, GY, GZ
      PRINT 116, KM
      K=K+JGA

31 IF (K-10) 30, 23, 23
23 CONTINUE
C   LUMPING OPERATION • BASED ON LACK OF AVAILBLE STORAGE
      LU=1
      CALL FAT X(XI, Z1, H1, 1)
      X(1,1)=X(1,1) + COST(1)
      CALL SHIFT
30 CONTINUE
      END FILE 49

500 CONTINUE
C   PRINTER SUMMARY
475 CONTINUE
      REWIND 49
      LC=LC+1
C   PAGE CONTROL DATA
      READ TAPE 49
      READ TAPE 49
      DO 301 I=1, NFIL
      IF(((I-1)/LC)*LC-I+1) 302, 303, 302
303 PRINT 300
      LC= NFIL/(NFIL/30+1)

```

```

PRINT 121
121 FORMAT(32X13HSTATE SUMMARY,/,11X5HNORTH,12X4HEAST,11X4HDOWN,
19X10HVELOCITIES,9X4HTIME,/,7X32HFILT REAL OBSR FILT REAL OBSR
2,32HFILT REAL FILT NORTH EAST DOWN,/)
302 READ TAPE 49
READ TAPE 49
READ TAPE 49,NA,KT
READ TAPE 49,R0,RR0,DNO,DE0,DD0
READ TAPE 49,XF,XDF,YF,YDF,ZF,ZDF
READ TAPE 49,AN
READ TAPE 49,GX,GY,GZ
DNO=DN0*R0
DE0=DE0*R0 $ KT=KT*2
DD0=DD0*R0
C PRINT CARTESIAN COORDINATES** FILTERED , TRUE, AND OBSERVED
301 PRINT 120,XF,GX,DNO,YF,GY,DE0,ZF,GZ,DD0,XDF,YDF,ZDF,KT
120 FORMAT(5X,3F6.1,6F5.2,3F6.3,I6,/)
REWIND 49 $ READ TAPE 49
READ TAPE 49
DO 304 I=1,NFIL
C MORE PAGE CONTROL
ITEST=((I-1)/LC)*LC-I+1
IF(ITEST)305,306,305
306 PRINT 300
PRINT 122
122 FORMAT(37X,18HOBSERVABLE SUMMARY,/,11X,5HRANGE,11X10HRANGE RATE,
116X,17HDIRECTION COSINES /42X5HNORTH,12X5HEAST ,12X,5HDOWN ,
26X4HTIME,/)
2 6X45HFILT REAL OBSR FILT OBSR FILT REAL OBSR ,
335H FILT REAL OBSR FILT REAL OBSR ,/)
305 READ TAPE 49
READ TAPE 49
READ TAPE 49,NA,KT
READ TAPE 49,R0,RR0,DNO,DE0,DD0
READ TAPE 49,XF,XDF,YF,YDF,ZF,ZDF
READ TAPE 49,AN

```

```

READ TAPE 49,GX,GY,GZ
RF=XF**2+YF**2+ZF**2
RF=SQRTF(RF)
DNF=XF/RF
DEF=YF/RF
DDF=ZF/RF
RRF= DEF*YDF + DDF*ZDF + DNF*XDF
RG=GX**2+GY**2 +GZ**2
RG=SQRTF(RG)
DNG=GX/RG
DEG=GY/RG $ KT=KT*2
DDG=GZ/RG

C OBSERVABLE SUMMARY PRINTED IN THIS ORDER RANGE,RANGE RATE,
C AND THREE DIRECTION COSINES. ,FILTERED , TRUE , AND OBSERVED.
C RANGE RATE TRUE NOT AVAILABLE ON INPUT TAPE
304 PRINT 123,RF,RG,RO,RRF,RR0,DNF,DNG,DNO,DEF,DEG,DE0,DDF,DDG,DD0,KT
123 FORMAT (5X,3F6.1,2X,2F5.3,2X,3F5.3,2X3F5.3,2X,3F5.3,I6,/)
REWIND 49
READ TAPE 49 $ TIN=0.
READ TAPE 49
DO 307 I=1,NFIL
C MORE PAGE CONTROL
ITEST=((I-1)/LC)*LC-I+1
IF(ITEST)308,309,308
309 PRINT 300
PRINT 124
124 FORMAT(40H
308 READ TAPE 4 9 TIME SMOOTH ERROR COST )
READ TAPE 49 , TIMA $ TIN=TIN-TIMA
READ TAPE 49,NA,KT,KS,AER,COSTA
READ TAPE 49,AN $KT=2*KT
READ TAPE 49,AN
READ TAPE 49,AN
READ TAPE 49,AN
150 FORMAT(2F10.3)
C TIMING AND CONSISTENCY DATA

```

```
PRINT 150 , TIMA , TIN
307 PRINT 125,KT,KS,AER,COSTA
125 FORMAT( 5X,2I6,2F10.4,/)
END
```

SUBROUTINE RUN1

```

C THIS SUBROUTINE HANDLES THE FIRST ITERATION ON EACH NEW PIECE OF
C DATA. THE INPUT IS X(N+1/N), P(N+1/N), AND Z(N+1). THE OUTPUT IS THE
C FIRST ITERATION ON X(N+1/N+1) AND P(N+1/N+1). THE COST C(N/N)* IS UP-
C DATED TO C(N/N+1) BEFORE THE FIRST AFTER PROCESSING AND THE COST OF THE FIRST
C ITERATION, C(N+1/N+1) IS COMPUTED AFTER PROCESSING.
COMMON PHI(8,8),DEL(8,8),Q(8,8),X10(8,10),P10(80,80),R(8),S10(8,8)
1,Y10(8),X(8,10),SQ(8,8),SR(8),Z(8,10),H(8,8,10),DELS(8,8),Z1(8,10)
2,H1(8,8,10),ZA(8,10),X1(8,10),X2(8,10),PB(80,80),NS,NW,NM,MAX,NU,
3 K,LSHOOT,LU,KMAX,COST1,COST2
4 ,RR(8,10)
5,ABC(8,8,10),BCD(8,8,10),NR(10),NSK(10)
DO 1 J=1,K
DO 1 I=1,NS
1 X1(I,J)=X2(I,J)
CALL F AT X (X1,Z1,H1,K)
DO 6 I=1,NM
6 COST1 =COST1+((Z(I,K)-Z1(I,K))**2)/RR(I,K)
PRINT 106,COST1
CALL KAL FIL (K)
106 FORMAT ( 9H COST1 = ,E10.3)
END

```

SUBROUTINE RUN2

```

C THIS SUBROUTINE HANDLES THE MECHANICS OF THE MAIN ITERATION LOOP ON
C X(N/N). THE BASIC DECISION FUNCTIONS ARE EXPRESSED IN JUMP AND LUMP.
C IF AN OVERSHOOT IS DETECTED THAT IS THE LAST COST IS GREATER THAN
C THE FIRST THEN THE STEP SIZE IS HALVED AND THE COST RE-EVALUATED,(JU=1).
C IF THE LEAST SQUARES FIT HAS NOT CONVERGED THE DATA IS REPROCESSED
C (JU=2)
C IF CONVERGENCE HAS OCCURRED AND A LUMPING OPERATION IS REQUIRED
C A FLAG IS SET AND THE LUMPING OPERATION TAKES PLACE THE
C NEXT TIME THE FULL MINIMIZATION PROCEDURE IS PERFORMED
C (JU=3, LU NOT EQUAL TO ZERO)
C IF NO FURTHER ITERATIONS ARE REQUIRED THEN(JU=4) AND EXIT RUN2.
C INITIALIZE ROUTINE
COMMON PHI(8,8),DEL(8,8),Q(8,8),X10(8,10),P10(80,80),R(8),S10(8,8)
1,Y10(8),X(8,10),SQ(8,8),SR(8),Z(8,10),H(8,8,10),DELS(8,8),Z1(8,10)
2,H1(8,8,10),ZA(8,10),X1(8,10),X2(8,10),PB(80,80),NS,NW,NM,MAX,NU,
3 K,LSHOOT,LU,KMAX,COST1,COST2
4 ,RR(8,10)
5,ABC(8,8,10),BCD(8,8,10),NR(10),NSK(10)
NIT =12
JU=0
KP=NS*K
12 DO 7 I=1,K
CALL F AT X (X2,Z1,H1,I)
7 CONTINUE
COST2=COST(K )
JU=JUMP(JU)
GO TO (9,14,24,20),JU
9 DO 11 J=1,K
DO 11 I=1,NS
11 X2(I,J)=(X1(I,J)+X2(I,J))*0.5
GO TO 12
24 CONTINUE

```

```

CALL AUG MENT ( P10, X10)
CALL AUG MENT (PB,X2)
K=K+1
14 DO 15 I=1,K
    IF (NIT) 2,3,2
2 CONTINUE
C MAIN MINIMIZATION
DC 15 J=1,NS
X1(J,I)=X2(J,I)
15 X2(J,I)=X10(J,I)
COST1=COST2
DO 16 I=1,KP
DO 16 J=I,KP
16 PB(I,J)=PB(J,I)=P10(J,I)
NIT=NIT-1
CALL RUN3
GO TO (12,12,30,20),JU
30 K=K-1 $ RETURN
3 CONTINUE
20 CONTINUE
COST1=COST2
CALL AUG MENT ( P10, X10)
CALL AUG MENT (PB,X2)
END

```


SUBROUTINE RUN3

```

C THIS SUBROUTINE IS THE BASIC ITERATION AND CONTAINS PROVISIONS FOR
C SENSING THE LUMPING REQUIREMENTS. WHEN LUMPING IS REQUIRED SHIFT IS
C CALLED
COMMON PHI(8,8),DEL(8,8),Q(8,8),X10(8,10),P10(80,80),R(8),S10(8,8)
1,Y10(8),X(8,10),SQ(8,8),SR(8),Z(8,10),H(8,8,10),DELS(8,8),Z1(8,10)
2,H1(8,8,10),ZA(8,10),X1(8,10),X2(8,10),PB(80,80),NS,NW,NM,MAX,NU,
3 K,LSHOOT,LU,KMAX,COST1,COST2
4 ,RR(8,10)
5,ABC(8,8,10),BCD(8,8,10),NR(10),NSK(10)
LU= LSHOOT
KS=K $LS=LU $IC=0
112 FORMAT ( 30H LUMPING OPERATION OCCURRED , E10.3 )
DO 21 I=1,KS
IB=I-IC*LS
25 CALL KAL FIL(IB)
IF (LU-I)24,23,24
23 CONTINUE
DO 26 ILU=1,LU
C EVALUATE NONLINEAR FUNCTIONS AT ESTIMATES
26 CALL FATX(X2,Z1,H1,ILU)
C ACCUMULATE LUMPED COST
X(1,1)=X(1,1) + COST(LU)
CALL SHIFT
IC=1
KP=K*NS $PRINT 112 ,X(1,1)
24 CONTINUE
21 CONTINUE
LU=
LSHOOT =0
END

```

SUBROUTINE INIT

```

C THIS ROUTINE INITIALIZES THE STATE VECTOR ACCORDING TO THE STATEMENT
C OF THE PROBLEM. IT ALSO CONVERTS CERTAIN PROBLEM STATEMENT MATRICES
C TO STANDARD FORM.
COMMON PHI(8,8),DEL(8,8),Q(8,8),X10(8,10),P10(80,80),R(8),S10(8,8)
1,Y10(8),X(8,10),SQ(8,8),SR(8),Z(8,10),H(8,8,10),DELS(8,8),Z1(8,10)
2,H1(8,8,10),ZA(8,10),X1(8,10),X2(8,10),PB(80,80),NS,NW,NM,MAX,NU,
3 K,LSHOOT,LU,KMAX,COST1,COST2
4 ,RR(8,10)
5,ABC(8,8,10),BCD(8,8,10),NR(10),NSK(10)

EPS = .0000001
CALL DECOM (P10,S10,NS,NP,EPS,80,8)
PRINT 113
DO 7 I=1,NS
7 PRINT 104,(S10(I,J),J=1,NP)
C END DECOMPOSE P10
NU=1220703125
DO 8 I=1,NP
8 CALL RNDEV63(NU,Y10(I))
PRINT 114
PRINT 104,(Y10(I),I=1,NP)
DO 9 I=1,NS
A=X10(I)
DO 10 J=1,NP
10 A=A+S10(I,J)*Y10(J)
9 X(I,1)=A
PRINT 115
PRINT 104,(X(I,1),I=1,NS)
C END GENERATOR START
CALL DECOM(Q,SQ,NW,NWR,EPS,8,8)
PRINT 116
DO 24 I=1,NW

```

```

24 PRINT 104,(SQ(I,J),J=1,NWR)
C END DECOMPOSE Q
DO 13 I=1,NM
13 SR(I)=SQRTF(R(I))
PRINT 117
PRINT 104,(SR(I),I=1,NM)
C END DECOMPOSE R
DO 15 I=1,NS
DO 15 K=1,NWR
A=0.
DO 14 J=1,NW
14 A=A+DEL(I,J)*SQ(J,K)
15 DELS(I,K)=A
PRINT 118 $ NW=NWR
DO 25 I=1,NS
25 PRINT 104,(DELS(I,J),J=1,NW)
DO 2 I=1,NS $ DO2 J=1,I $ A=0. $ DO 1K0 =1,NW
1 A=A + DELS(I,KO) * DELS(J,KO)
2 Q(I,J)=Q(J,I) =A
C ADD GAUSS3 TO PACK
DO 3 I=1,NW $ DO 3 J=1,I $ A=0.$ DO 4 L=1,NS
4 A=A+DELS(L,I)*DELS(L,J)
3 DEL(I,J)=DEL(J,I)=A
IF(NW-1)27,26,27
29 PRINT 118 $ RETURN
27 CALL GAUSS3(NW,1,E-8,DEL,SQ,KER,8)
IF(KER.EQ.2)29,28
26 SQ(1,1)=1./DEL(1,1) $GOTO28
28 DO 5 I=1,NW $DO 5 J=1,NS $ A=0. $ DO 6 L=1,NW
6 A=A+ DELS(J,L)*SQ(L,I)
5 DEL(I,J)=A
DO 12 I=1,NW $ DO 12 J=1,NS $ A=0. $ DO 11 L=1,NS
11 A=A+DEL(I,L)*PHI(L,J)
12 SQ(I,J)=A
104 FORMAT (1H ,8E10.3)
113 FORMAT ( 20H SQUARE ROOT OF P10 )

```

```

114 FORMAT ( 30H RANDOM VECTOR FROM N(0,I) HAT)
115 FORMAT ( 30H INITIAL VALUE OF STATE VECTOR)
116 FORMAT ( 20H SQUARE ROOT OF Q )
117 FORMAT ( 20H SQUARE ROOT OF R )
118 FORMAT ( 30H NORMALIZED INPUT MATRIX DELS )
PRINT 200 $ CALL TIME
200 FORMAT(30X,30HEXIT SUB INIT
      )
END

```

SUBROUTINE READ

```

C THIS ROUTINE READS IN AND SUMMARIZES THE PROBLEM STATEMENT DATA.
COMMON PHI(8,8),DEL(8,8),Q(8,8),X10(8,10),P10(80,80),R(8),S10(8,8)
1,Y10(8),X(8,10),SQ(8,8),SR(8),Z(8,10),H(8,8,10),DFLS(8,8),Z1(8,10)
2,H1(8,8,10),ZA(8,10),X1(8,10),X2(8,10),PB(80,80),NS,NW,NM,MAX,NU,
3 K,LSHOOT,LU,KMAX,COST1,COST2
4 ,RR(8,10)
5,ABC(8,8,10),BCD(8,8,10),NR(10),NSK(10)
READ 101,NS
PRINT 102,NS
PRINT 103
DO 1 I=1,NS
READ 104,(PHI(I,J),J=1,NS)
1 PRINT 104,(PHI(I,J),J=1,NS)
READ 101,NW
PRINT 105,NW
PRINT 106
DO 2 I=1,NS
READ 104,(DEL(I,J),J=1,NW)
2 PRINT 104,(DEL(I,J),J=1,NW)
PRINT 107
DO 3 I=1,NW
READ 104,(Q(I,J),J=1,NW)
3 PRINT 104,(Q(I,J),J=1,NW)
PRINT 108
READ 104,(X10(I,1),I=1,NS)
PRINT 104,(X10(I,1),I=1,NS)
PRINT 109
DO 4 I=1,NS
READ 104,(P10(I,J),J=1,NS)
4 PRINT 104,(P10(I,J),J=1,NS)
READ 101,NM
PRINT 111,NM

```

```

PRINT 110
READ 104,(R(I),I=1,NM)
PRINT 104,(R(I),I=1,NM)
READ 101,MAX
PRINT 112, MAX

C END INPUT ROUTINE
101 FORMAT(I2)
102 FORMAT ( 39 H SIGNAL GENERATING SYSTEM OF DIMENSION ,I2)
103 FORMAT ( 30H SYSTEM TRANSITION MATRIX PHI )
104 FORMAT (1H ,8E10.3)
105 FORMAT ( 47H THE NUMBER OF INDEPENDENT WHITE NOISE INPUTS = ,I2)
106 FORMAT ( 28H SYSTEM INPUT MATRIX DEL )
107 FORMAT ( 28H INPUT NOISE COVARIANCE Q )
108 FORMAT ( 40H APRIORI ESTIMATE OF FIRST STATE VECTOR )
109 FORMAT ( 30H APRIORI COVARIANCE MATRIX )
110 FORMAT ( 27H MEASUREMENT VARIANCE R )
111 FORMAT ( 25H NUMBER OF MEASUREMENTS =,I2)
112 FORMAT ( 33H NUMBER OF STAGES OF OPERATION = ,I2 )
PRINT 200 $ CALL TIME
200 FORMAT(30X,30HEXIT SUB READ
      )
END

```

SUBROUTINE KAL FIL (IST)

```

COMMON PHI(8,8),DEL(8,8),Q(8,8),X10(8,10),P10(80,80),R(8),S10(8,8)
1,Y10(8),X(8,10),SQ(8,8),SR(8),Z(8,10),H(8,8,10),DELS(8,8),Z1(8,10)
2,H1(8,8,10),ZA(8,10),X1(8,10),X2(8,10),PB(80,80),NS,NW,NM,MAX,NU,
3 K,LSHOOT,LU,KMAX,COST1,COST2
4 ,RR(8,10)
5,ABC(8,8,10),BCD(8,8,10),NR(10),NSK(10)
C THIS SUBROUTINE PERFORMS STANDARD KALMAN LINEAR FILTERING ON A MULTI
C STAGE STATE VECTOR AND AN OVERALL COVARIANCE. THE ARGUMENT IST INDICATES
C THE TIME OF THE OBSERVATION HOWEVER ALL STAGES ARE UPDATED. THIS ROUTINE
C HAS BUILT IN FEATURE OF INCLUDING SECOND ORDER TERMS JUST PRIOR TO A
C LUMPING OPERATION.
C DIMENSION G(80)
KP =K*NS
DO 5 IA=1,NM
DO 2 IB=1,KP $A=0.
DO1 IC=1,NS $IE=IC+NS*(IST-1)
1 A=A+ PB (IB,IE)*H1(IA,IC,IST)
2 G(IB)=A
E =Z(IA,IST)-Z1(IA,IST) $ B=RR(IA,IST)
IF(LU-IST)6,7,7
6 B=B+ZA(IA,IST)
7 CONTINUE
DO3 IB=1,NS $ IC=IB+NS*(IST-1)
B=B+G(IC)*H1(IA,IB,IST)
3 E=E-H1(IA,IB,IST)*(X2(IB,IST)-X1(IB,IST))
C WILD DATA TEST
C NOT USED
GO TO 8
CHISQ = E*E/B
IF(CHISQ-10.) 8,9,9
9 RR(IA,IST)=RR(IA,IST)*10000.* CHISQ
E=0.

```

```

X(1,2)=X(1,2) +1.
PRINT 101,IA,IST
101 FORMAT ( 28H OUT OF BOUNDS DATA   OBS   , I2, 9H  STAGE   ,I2)
8 CONTINUE
DO 5 IB=1,KP $C=G(IB)/B
IS=(IB-1)/NS+1$IR=IB-IS*NS+NS
DO 4 IC=IB,KP
PB (IB,IC)=PB (IC,IB)=PR (IC,IB)-C*G(IC)
5 X2(IR,IS) =X2(IR,IS) +C*E
END

```


FUNCTION LUMP (I)

```

C THIS ROUTINE TEST TO SEE IF THE NONLINEAR NOISE IS LESS THAN
C A CERTAIN RATIO ,CN, OF THE NATURAL NOISE
COMMON PHI(8,8),DEL(8,8),Q(8,8),X10(8,10),P10(80,80),R(8),S10(8,8)
1,Y10(8),X(8,10),SQ(8,8),SR(8),Z(8,10),H(8,8,10),DELS(8,8),Z1(8,10)
2,H1(8,8,10),ZA(8,10),X1(8,10),X2(8,10),PB(80,80),NS,NW,NM,MAX,NU,
3 K,LSHOOT,LU,KMAX,COST1,COST2
4 ,RR(8,10)
5,ABC(8,8,10),BCD(8,8,10),NR(10),NSK(10)
DIMENSION IT(12),XG(12),YG(12)
1,B(8),PT(8,8),R2(8,8)
CN=10.
C EXTRACT THE COVARIANCE MATRIX OF AN INDIVIDUAL STATE
C ESTIMATE FROM THE OVERALL COVARIANCE MATRIX
DO 15 L=1,K $LN=L*NS-NS$DO 14 I=1,NS$IA=I+LN$DO 14J=1,NS$JA=J+LN
14 PT(I,J)=PT(J,I)=PB(IA,JA)
LUMP =L-1
CALL SEC ORDER (B,R2,PT,NM,NS,8,8,X2,L)
B IS THE ESTIMATE BIAS
R2 IS THE MATRIX OF
R2 IS THE COVARIANCE MATRIX OF SECOND ORDER NOISE
PT IS THE INDIVIDUAL ESTIMATE COVARIANCE
X2 IS THE WHOLE COLLECTION OF ESTIMATES
L INDICATES THE ESTIMATE UNDER CONSIDERATION
DO 20 J=1,NM $IF(CN*R2(J,J)-RR(J,L))20,16 ,16
BASIC TEST
C LUMP TAKES ON A VALUE EQUAL TO THE THE NUMBER OF ESTIMATEF
C WHICH PASS THE TEST
16 IF(L-1)21,22,21
21 RETURN
22 IF(K-8)21,20,21
20 CONTINUE
C SET UP TRANSFER OF NONLINEAR TERMS TO BASIC FILTER

```

```
DO 18 J=1,NM $ZA(J,L)=R2(J,J)
18 Z1(J,L)=Z1(J,L)-B(J)
15 CONTINUE
LUMP=K
END
```

FUNCTION JUMP(JU)

```

C THIS ROUTINE TEST FOR CONVERGENCE AND CALLS THE
C ROUTINE LUMP TO TEST FOR LUMPING
  COMMON PHI(8,8),DEL(8,8),Q(8,8),X10(8,10),P10(80,80),R(8),S10(8,8)
  1,Y10(8),X(8,10),SQ(8,8),SR(8),Z(8,10),H(8,8,10),DELS(8,8),Z1(8,10)
  2,H1(8,8,10),ZA(8,10),X1(8,10),X2(8,10),PB(80,80),NS,NW,NM,MAX,NU,
  3 K,LSHOOT,LU,KMAX,COST1,COST2
  4 ,RR(8,10)
  5,ABC(8,8,10),BCD(8,8,10),NR(10),NSK(10)
C CONVERGENCE TEST PARAMETERS
  DIMENSION DELC(10),CLIM(10)
  DATA
  1 (DELC(J),J=1,10)=1.6,5.2,9.,14.,18.,22.,26.,30.,34.,38.),
  2 ((CLIM(J),J=1,10)= 11.,18.,25.,31.,38.,44.,50.,56.,62.,78.)
  EPS1=.000000001
  IF(JU-3)50,4,50
  50 A=0.$ DO 35 I=1,NS $ DO 35 J=1,K
  C RELATIVE CHANGE IN ESTIMATES TEST
  35 A= MAX1F(ABSF((X2(I,J) -X1( I,J))/X2(I,J)),A)
  IF(A-EPS1)5,5,6
C INSIGNIFICANT DIFFERENCE TEST
C OVERTHOOT TEST
  6 IF(COST1-COST2+DELC(K)*.001) 1,1,7
  7 IF(JU-2)2,8,8
C POSSIBLE MINIMUM COST TEST
  8 IF (COST1-CLIM(K))9,9,2
  9 IF(COST2+DELC(K)-COST1) 2,5,5
C COMPLETED CONVERGENCE TEST
  5 LU=LUMP(IDUM )
  LSHOOT =LU
  IF(LU)4,4,3
  1 JUMP=1$PRINT 101,COST2,A $RETURN
101 FORMAT(40H OVERTHOOT COST2=

```

```

1 ,F10.2,10H REL CHANG ,F10.7)
2 JUMP=2$PRINT 102,COST2,A $RET(JRN
102 FORMAT(40H REPROCESS
1 ,F10.2,10H REL CHANG ,F10.7)
3 JUMP=4$PRINT 103,COST2,A $RETURN
103 FORMAT(40H REPROCESS LAST STAGE AND LUMP COST2=
1 ,F10.2,10H REL CHANG ,F10.7)
4 JUMP=4$PRINT 104,COST2,A $RETURN
104 FORMAT(40H COMPLETED PROCESS COST2=
1 ,F10.2,10H REL CHANG ,F10.7)
END

```

FUNCTION COST(KA)

```
C THIS ROUTINE EVALUATES THE SUM OF THE SQUARES OF THE RESIDUALS
C WEIGHED IN PROPORTION TO THE PSEUDO INVERSE OF THE COVARIANCE OF THE
C ASSOCIATED NOISE
COMMON PHI(8,8),DEL(8,8),Q(8,8),X10(8,10),P10(80,80),R(8),S10(8,8)
1,Y10(8),X(8,10),SQ(8,8),SR(8),Z(8,10),H(8,8,10),DELS(8,8),Z1(8,10)
2,H1(8,8,10),ZA(8,10),X1(8,10),X2(8,10),PB(80,80),NS,NW,NM,MAX,NU,
3 K,LSHOOT,LU,KMAX,COST1,COST2
4 ,RR(8,10)
5,ABC(8,8,10),BCD(8,8,10),NR(10),NSK(10)
COST =0.
DO 3 I=1,NU $ A=0.
DO 2 J=1,NS
2 A=A+S10(I,J)*(X10(J,1)-X2(J,1))
3 COST=COST+A**2
DO 1 I=1,NM
1 COST=COST+(Z(I,1)-Z1(I,1))**2/RR(I,1)
DO 6 I=2,KA $ IA=I-1
NRR=NR(IA)
DO 5 J=1,NRR
A=0.
DO 4 L=1,NS
4 A=A+ABC(J,L,IA)*X2(L,I)-BCD(J,L,IA)*X2(L,IA)
5 COST=COST+A*A
DO 6 J=1,NM
6 COST=COST +(Z(J,I)-Z1(J,I))**2/RR(J,I)
END
```

IDENT ISWF

ENTRY ISWF
SLJ **
LDA ISWF
ARS 24
INA 1
SAU ISWF
SLJ 1 UP
ENA 0
SLJ ISWF
ENA 1
SLJ ISWF
END

ISWF

UP

SW KEY 1 IS DOWN
SW KEY 1 IS UP

SUBROUTINE AUGMENT(P, XA)

```
C THIS SUBROUTINE COMPUTES THE REQUIRED ADDITIONS TO THE COVARIANCE
C MATRIX AND THE STATE ESTIMATE VECTOR WHENEVER THE NUMBER OF STAGES
C CARRIED INCREASES BY 1.
  DIMENSION P(80,80),XA(8,10)
  COMMON PHI(8,8),DEL(8,8),G(8,8),X10(8,10),P10(80,80),R(8),S10(8,8)
  1,Y10(8),X(8,10),SQ(8,8),SR(8),Z(8,10),H(8,8,10),DELS(8,8),Z1(8,10)
  2,H1(8,8,10),ZA(8,10),X1(8,10),X2(8,10),PB(80,80),NS,NM,MAX,NU,
  3 K,LSHOOT,LU,KMAX,COST1,COST2
  4 ,RR(8,10)
  5,ABC(8,8,10),BCD(8,8,10),NR(10),NSK(10)
  DO 2 I=1,K
  DO 2 J=1,NS $JA=J+NS*K
  DO 2 L=1,NS $ LB=L+NS*I $LA=LB-NS
  A=0.
  DO 1 N=1,NS $ NA=N+NS*(K-1)
  1 A=A+PHI(J,N)*P(NA,LA)
  2 P(JA,LA)=P(LA,JA)=A
  DO 4 J=1,NS$JA=J+NS*K
  DO 4 L=J,NS$LA=L+NS*K
  A=Q(J,L)
  DO 3 N=1,NS $ NA=N+NS*(K-1)
  3 A=A+P(JA,NA)*PHI(L,N)
  4 P(JA,LA)=P(LA,JA)=A
  DO 5 I=1,NS
  A=0.
  DO 6 L=1,NS
  6 A=A+XA(L,K)*PHI(I,L)
  5 XA(I,K+1)=A
  END
```

SUBROUTINE SHIFT

```

C THIS SUBROUTINE ESSENTIALLY SHIFTS THE TIME AXIS SO THE OLDEST
C STAGE CURRENTLY (AFTER LUMPING) IS CALLED 1
C ALL DATA, MATRICES OF PARTIALS, STATE ESTIMATES, COVARIANCES ARE
C SHIFTED. THE OVERALL COVARIANCE MATRIX AND ASSOCIATED ESTIMATE AFTER
C THE DATA FROM THOSE STAGES TO BE LUMPED HAS BEEN FILTERED, BECOME
C THE NEW APRIORI DATA.
C DIMENSION B(8,8)
C COMMON PHI(8,8),DEL(8,8),Q(8,8),X10(8,10),P10(80,80),K(8),S10(8,8)
1,Y10(8),X(8,10),SQ(8,8),SR(8),Z(8,10),H(8,8,10),DELS(8,8),Z1(8,10)
2,H1(8,8,10),ZA(8,10),X1(8,10),X2(8,10),PB(80,80),NS,NW,NM,MAX,NU,
3 K,LSHOOT,LU,KMAX,COST1,COST2
4 ,RR(8,10)
5,ABC(8,8,10),BCD(8,8,10),NR(10),NSK(10)
IF(LU)12,13,12
12 K=K-LU $ KP=K*NS $ KLM=NM*LU
KM=NM*K $ KLU=NS*LU
DO 2 I=1,KP
DO 2 J=1,KP
2 P10(I,J)=P10(J,I)=PB(I,J)=PB(J,I)=PB(I+KLU,J+KLU)
DO 1 J1=1,K
DO 1 J2=1,NM
RR(J2,J1)=RR(J2,J1+LU)
Z1(J2,J1)=Z1(J2,J1+LU)
1 Z(J2,J1)=Z(J2,J1+LU)
DO 4 J1=1,K
DO 5 J2=1,NM
DO 5 J3=1,NS
5 H1(J2,J3,J1)=H1(J2,J3,J1+LU)
DO 4 J4=1,NS
NRR=NR(J1)
DO 14 J3=1,NRR
ABC(J3,J4,J1)=ABC(J3,J4,J1+LU)

```



```

14 BCD(J3,J4,J1)=BCD(J3,J4,J1+LU)
   X10(J4,J1)=X2(J4,J1) =X2(J4,J1+LU)
   4 X1(J4,J1)=X1(J4,J1+LU)
   DO 15 J1=1,K $ NSK(J1)=NSK(J1+LU)
15 NR(J1)=NR(J1+LU)
13 CALL DECOM A(P10,DELS,S10,NS,NU,1.E-8,80,8)
   IF(NU-NS)7,6,7
   7 DO 8 I=1,NU $ DO 8 J=1,I $ A=0. $ DO 9 L=1,NS
   9 A=A+DELS(L,I)*DELS(L,J)
   8 S10(I,J)=S10(J,I)=A
   CALL GAUSS3(NU,1.E-8,S10,B,KLP,8)
   DO 10 I=1,NU $ DO 10 J=1,NS $ A=0. $ DO 11 L=1,NU
11 A=A+B(I,L)*DELS(J,L)
10 S10(I,J)=A
   6 CONTINUE
   END

```

SUBROUTINE FAT X(X,Z,H,K)

```
C THIS ROUTINE EVALUATE THE NONLINEAR MEASUREMENT FUNCTIONS
C AND THE FIRST PARTIAL DERIVATIVES
C THIS ROUTINE MUST BE REWRITTEN FOR EACH NEW MODEL INVOLVING
C DIFFERENT MEASUREMENT FUNCTIONS
  DIMENSION X(8,10),H(8,8,10),Z(8,10)
  RS=X(1,K)**2 +X(3,K)**2 + X(5,K)**2
  Z(1,K)=R = SQRTF(RS)
  Z(2,K)=(X(1,K)*X(2,K)+X(3,K)*X(4,K)+X(5,K)*X(6,K))/R
  Z(3,K)=X(1,K)/R
  Z(4,K)=X(3,K)/R
  Z(5,K)=X(5,K)/R
  DO 1 I=1,5
  DO 1 J=1,6
1 H(I,J,K)=0.
  H(1,1,K)=X(1,K)/R
  H(1,3,K)=X(3,K)/R
  H(1,5,K)=X(5,K)/R
  H(2,2,K)=Z(3,K)
  H(2,4,K)=Z(4,K)
  H(2,1,K)=(X(2,K)*R-Z(2,K)*X(1,K))/RS
  H(2,3,K)=(X(4,K)*R-Z(2,K)*X(3,K))/RS
  H(2,5,K)=(X(6,K)*R-Z(2,K)*X(5,K))/RS
  H(2,6,K)=Z(5,K)
  H(3,1,K)=1./R-X(1,K)*H(1,1,K)/RS
  H(3,3,K)= -X(1,K)*H(1,3,K)/RS
  H(3,5,K)= -X(1,K)*H(1,5,K)/RS
  H(4,1,K)=-X(3,K)*H(1,1,K)/RS
  H(4,3,K)=1./R -X(3,K)*H(1,3,K)/RS
  H(4,5,K)= -X(3,K)*H(1,5,K)/RS
  H(5,1,K)= -X(5,K)*H(1,1,K)/RS
  H(5,3,K)= -X(5,K)*H(1,3,K)/RS
  H(5,5,K)=1./R -X(5,K)*H(1,5,K)/RS
END
```

```

SUBROUTINE SEC ORDER (B,R2,PT,NM,NS,NMD,NSD,X,IST)

C THIS ROUTINE EVALUATES THE MATRIX OF SECOND PARTIALS AND
C COMPUTES THE SECOND ORDER TERMS IN THE EXPANSION OF THE
C NONLINEAR MEASUREMENT FUNCTIONS
C THIS ROUTINE MUST BE WRITTEN FOR EACH NEW SET OF
C MEASUREMENT FUNCTIONS
DIMENSION A(6,6,5),PT(NSD,NSD),R1(6,6,6),B(NMD),R2(NMD,NMD)
1,X(8,10)
DO 1 I=1,6 $ DO 1 J=1,6 $ DO 1 K=1,5
1 A(I,J,K)=0.
K=IST
RS=X(1,K)**2+X(3,K)**2+X(5,K)**2
R=SQRTF(RS)
RC=R*RS
RF=R*RC
H1=-X(1,K)/RC
H3=-X(3,K)/RC
H5=-X(5,K)/RC
RDOT=(X(1,K)*X(2,K)+X(3,K)*X(4,K)+X(5,K)*X(6,K))/R
A(1,1,2)=2.*H1*X(2,K)-RDOT/RS+3.*H1*H1*RDOT*RS
A(3,3,2)=2.*H3*X(4,K)-RDOT/RS+3.*H3*H3*RDOT*RS
A(5,5,2)=2.*H5*X(6,K)-RDOT/RS+3.*H5*H5*RDOT*RS
A(1,2,2)=A(2,1,2)=1./R+H1*X(1,K)
A(3,4,2)=A(4,3,2)=1./R+H3*X(3,K)
A(5,6,2)=A(6,5,2)=1./R+H5*X(5,K)
A(1,3,2)=A(3,1,2)=H1*X(4,K)+H3*X(2,K)+3.*RDOT*H1*H3*RS
A(1,5,2)=A(5,1,2)=H1*X(6,K)+H5*X(2,K)+3.*RDOT*H1*H5*RS
A(3,5,2)=A(5,3,2)=H3*X(6,K)+H5*X(4,K)+3.*RDOT*H3*H5*RS
A(1,4,2)=A(4,1,2)=A(2,3,2)=A(3,2,2)=H1*X(3,K)
A(1,6,2)=A(6,1,2)=A(2,5,2)=A(5,2,2)=H5*X(1,K)
A(3,6,2)=A(6,3,2)=A(4,5,2)=A(5,4,2)=H3*X(5,K)
A(1,1,1)=1./R+X(1,K)*H1
A(3,3,1)=1./R+X(3,K)*H3

```

```

A(5,5,1)=1./R+X(5,K)*H3
A(1,3,1)=A(3,1,1)=H1*X(3,K)
A(1,5,1)=A(5,1,1)=H5*X(1,K)
A(5,3,1)=A(3,5,1)=H3*X(5,K)
A(1,1,3)=(X(1,K)**3/RF-H1)*3.
A(3,3,4)=(X(3,K)**3/RF-H3)*3.
A(5,5,5)=(X(5,K)**3/RF-H5)*3.
A(1,1,4)=A(1,3,3)=A(3,1,3)=3.*H1*H1*H3*RS*RS-H1
A(1,1,5)=A(1,5,3)=A(5,1,3)=3.*H1*H1*H5*RS*RS-H1
A(3,3,3)=A(3,1,4)=A(1,3,4)=3.*H3*H3*H1*RS*RS-H3
A(3,3,5)=A(3,5,4)=A(5,3,4)=3.*H3*H3*H5*RS*RS-H3
A(5,5,3)=A(5,1,5)=A(1,5,5)=3.*H5*H5*H1*RS*RS-H5
A(5,5,4)=A(5,3,5)=A(3,5,5)=3.*H5*H5*H3*RS*RS-H5
A(1,3,5)=A(1,5,4)=A(3,1,5)=A(3,5,3)=A(5,1,4)=A(5,3,3)=
13.*H1*H5*H3*RS*RS
DO 2 I=1,NM
B(I)=0.
DO 2 IA=1,NS
DO 3 IB=1,NS
C=0.
DO 4 IC=1,NS
4 C=C+A(IA,IC,I)*PT(IB,IC)
3 R1(IA,IB,I)=C
2 B(I)=B(I)+R1(IA,IA,I)
DO 5 I=1,NM
DO 5 J=1,I
C=0.
DO 6 IA=I,NS
DO 6 IB=1,NS
6 C=C+R1(IA,IB,I)*R1(IB,IA,J)
5 R2(I,J)=R2(J,I)=C
END

```

```

SUBROUTINE SKIP DATA
C THIS ROUTINE COMPUTES THE WEIGHTING MATICES FOR THE CASE
C WHERE THE DATA DO NOT ARRIVE AT EVEN INTERVALS
C THE EFFECTIVE PHI BECOMES THE BASIC INTERVAL PHI RAISED TO THE
C NTH POWER. THE EFFECTIVE Q IS COMPUTED BY A DIFFERENCE
C EQUATION EXERCISED N-1 TIMES .
C THE DIFFERENCE EQUATION IS
C Q+PHI*QC*PHI(TRANS) GOES TO QC
C WHERE Q IS THE SINGLE-INTERVAL Q
C AND QC IS THE MULTI-INTERVAL Q
COMMON PHI(8,8),DEL(8,8),Q(8,8),X10(8,10),P10(80,80),R(8),S10(8,8)
1,Y10(8),X(8,10),SQ(8,8),SR(8),Z(8,10),H(8,8,10),DELS(8,8),Z1(8,10)
2,H1(8,8,10),ZA(8,10),X1(8,10),X2(8,10),PB(80,80),NS,NW,NM,MAX,NU,
3 K,LSHOOT,LU,KMAX,COST1,COST2
4 ,RR(8,10)
5,ABC(8,8,10),BCD(8,8,10),NR(10),NSK(10)
DIMENSION QS(8,8),T(8,8)
DO 1 I=1,NS$DO 1 J=I,NS
1 QS(I,J)=QS(J,I)=Q(I,J)
NSKK=NSK(K)
DO 5 I=2,NSKK $ DO 3 J=1,NS $ DO 3 L=1,NS $ AC=0. $ DO 4 M=1,NS
4 AC=AC+PHI(J,M)*QS(M,L)
3 T(J,L)=AC $ DO 5 J=1,NS$ DO 5 L=J,NS$AC=Q(J,L)$DO 6 M=1,NS
6 AC=AC+T(J,M)*PHI(L,M)
5 QS(J,L)=QS(L,J)=AC
CALL DECOMA(QS,T,NS,NR(K),.0000001,8,8)
NRR=NR(K)
DO 2 I=1,NS $ DO 2 J=1,NRR
NSKK=NSK(K)
2 ABC(I,J,K)=BCD(I,J,K)=T(I,J) $ DO 9 IA=1,NSKK
NRR=NR(K)
DO 7 I=1,NS $ DO 7 J=1,NRR $ AC=0. $ DO 8 L=1,NS
8 AC=AC+PHI(L,I)*BCD(J,L,K)
7 T(I,J)=AC$ DO 9 J=1,NRR $ DO 9 I=1,NS
9 BCD(I,J,K)=T(I,J)
END

```

SUBROUTINE GAUSS3 (N,EP,A,X,KER,M)

000000

```

DIMENSION A(M,M),X(M,M)
DO 1 I=1,N
DO 1 J=1,N
1 X(I,J)=0.
DO 2 K=1,N
2 X(K,K)=1.
10 DO 34 L=1,N
KP=0
Z=0.
DO 12 K=L,N
IF(Z.GE.ABSF(A(K,L)))12,11
11 Z=ABSF(A(K,L))
KP=K
12 CONTINUE
IF(L.GE.KP)20,13
13 DO 14 J=L,N
Z=A(L,J)
A(L,J)=A(KP,J)
14 A(KP,J)=Z
DO 15 J=1,N
Z=X(L,J)
X(L,J)=X(KP,J)
15 X(KP,J)=Z
20 IF(ABSF(A(L,L)).LE.EP)50,30
30 IF(L.GE.N)34,31
31 LPI=L+1
DO 36 K=LPI,N
IF(A(K,L).EQ.0)36,32
32 RATIO=A(K,L)/A(L,L)
DO 33 J=LPI,N
33 A(K,J)=A(K,J)-RATIO*A(L,J)
DO 35 J=1,N
```

```

35 X(K,J)=X(K,J)-RATIO*X(L,J)
36 CONTINUE
34 CONTINUE
40 DO 43 I=1,N
   I1=N+1-I
   DO 43 J=1,N
     S=0.
     IF(I1.GE.N)43,41
     41 IIP1=I1+1
     DO 42 K=IIP1,N
       S=S+A(I1,K)*X(K,J)
     42 X(I1,J)=(X(I1,J)-S)/A(I1,I1)
     KER=1
   RETURN
50 KER=2
   END

```

```

000330
000340
000350
000360
000370
000380
000390
000400
000410
000420
000430
000440
000450
000460
000470

```

SUBROUTINE DECOM A (QA,B,C,NS,NR,EPS,N1,N2)

```
DIMENSIONQA(N1,N1),B(N2,N2),N(25),Q(25,25),C(N2,N2)
DO 9 I=1,NS
DO 8 J=1,NS
  B(I,J) = 0.
  C(I,J)=0.
8  Q(I,J) =QA(I,J)
9  C(I,I)=1.
  R=0.
DO 1 I=1,NS
  N(I)=I
DO17 I=1,NS $ NR=NS
  A1=R $ JT=0
DO 2 J=I,NS
  A=Q(N(J),N(J))
  IF (A+R)12,13,13
13 IF (A-A1)2,2,3
3  A1=A $ JS=N(J)$JT=J
  2 CONTINUE
  IF (I-1) 15,14,15
14 R=EPS*A1
15 IF(JT)6,5,6
  5 NR=I-1 $ RETURN
12 NR=0 $RETURN
6  N(JT)=N(I) $ N(I)=JS $ IA=I+1
  B(JS,I)=SQRTF(A1)
  A=B(JS,I)
  A2= 1./A
DO 7 J=IA,NS
7  B(N(J),I)=Q(N(J),N(I))*A2
DO 16 J=1,I
16 C(N(I),N(J))=C(N(I),N(J))*A2
DO17 J=IA,NS $ DO 4 K=J,NS
```



```
4 Q(N(K),N(J))=Q(N(J),N(K))=Q(N(K),N(J))-B(N(J),I)*B(N(K),I)
DO 17 K=1,I
17 C(N(J),N(K))
1 C(N(J),N(K)) =C(N(J),N(K)) -C(N(I),N(K))* B(N(J),I)
NR=NS
END
```

SUBROUTINE DECOM (QA,B,NS,NR,EPS,N1,N2)

```
DIMENSIONQA(N1,N1),B(N2,N2),N(25),Q(25,25)
DO 8 I=1,NS
DO 8 J=1,NS
  B(I,J) = 0.
  8 Q(I,J) =QA(I,J)
  R=0.
DO 1 I=1,NS
  1 N(I)=I
DO 4 I=1,NS $ NR=NS
  A1=R $ JT=0
DO 2 J=I,NS
  A=Q(N(J),N(J))
  IF(A+R)12,13,13
  13 IF(A-A1) 2,2,3
  3 A1=A $ JS=N(J)$JT=J
  2 CONTINUE
  IF(I-1)15,14,15
  14 R=EPS*A1
  15 IF(JT)6,5,6
  5 NR=I-1 $ RETURN
  12 NR=0 $RETURN
  6 N(JT)=N(I) $ N(I)=JS $ IA=I+1
  B(JS,I)=SORTF(A1)
  A=B(JS,I)
DO 7 J=IA,NS
  7 B(N(J),I)=Q(N(J),N(I))/A
DO 4 J=IA,NS $ DO 4 K=J,NS
  4 Q(N(K),N(J))=Q(N(J),N(K))-B(N(J),I)*B(N(K),I)
  NR=NS
  END
```

SR TIME

TIME

IDENT

ENTRY	TIME
SLJ	**
ENQ	0
LAC	=00
SCL	=04000000000000000000
DVI	=074
STG	=SIT3
ENQ	0
DVI	=074
STA	=SIT1
STG	=SIT2
CALL	TIMO
ZRO	IT1
ZRO	IT2
ZRO	IT3
SLJ	TIME
END	

TIME

7

+

+

SUBROUTINE TIMO(IT1,IT2,IT3)

```
3 FORMAT(26H0 INITIAL TIME IN SECONDS , F10.4 )
5 FORMAT(28H0 TIME DIFFERENCE IN SECONDS , F10.4 )
DECT=FLOATF(IT1
1      ) *60.+FLOATF(IT2) +FLOATF(IT3)/ 60.
1 IF (XX-37.)1,2,1
1 XX=37.
DECTO =0.
PRINT3,DECTO
DECTO =DECT
GO TO 4
2 DECTO =DECT-DECTO
PRINT 5, DECTO
WRITE TAPE 49, DECTO
DECTO=DECT
4 CONTINUE
END
```

IDENT RNDEV63

RNDEV63
 ENTRY RNDEV63
 SLJ **
 SIU 1 EXIT
 LIU 1 RNDEV63
 LDA 1 0
 SAL EXIT-1
 ARS 24
 SAU RDVLP
 SAL RDVLP+1
 INI 1
 SIL 1 EXIT
 ENA
 STA ERASE
 ENI 1 15
 LDA **
 MUI FIVE13
 SCL RDVCONS+1
 STA **
 RAD ERASE
 IJP 1 RDVLP
 SUB RDVCONS
 STA ERASE
 AJP 2 *+1
 LAC ERASE
 SCA 1 2 57B
 LRS 47
 ENA 1 0
 LLS 36
 SSK ERASE
 SCM RDVCONS+3
 SCM RDVCONS+3
 FMU RDVCONS+2

000 10
 0000 20
 0000 30
 0000* 40
 0000 5
 00000060
 00000 70
 00000080
 00000090
 0000010
 00000110
 0000012
 00000130
 00000140
 00000150
 00000160
 00000170
 00000180
 00000190
 00000200
 00000210
 00000220
 00000230
 00000240
 00000250
 00000260
 0000027
 00000280
 00000290
 00000300
 00000310
 00000320

SAVE INDEX
 SET SECOND ARG R
 SET FIRST ARG NUNIF
 SET FIRST ARG NUNIF
 SET EXIT
 **=ADDRESS OF INPUT ARG
 REDUCE MOD2 TO THE 43RD
 **=ADDRESS OF INPUT ARG
 SUBTRACT 2 TO THE 46TH
 SET FLAG
 COMPLEMENT IF NEG
 FLOAT A
 CHECK FOR NEGATIVE DEVIATE
 DIVIDE BY STANDARD DEVIATION

00000330
00000340
00000350
00000360
00000370
0000 380
0000 390
00000400
0000 41

** = ADDRESS OF SECOND ARG R

**
**
**
1
1220703125
2000000000000000
7600000000000000
1724673317272054
-

1

STA
ENI
SLJ
BSS
DEC
OCT
OCT
OCT
OCT
END

-
EXIT
ERASE
FIVE13
RDVCONS

INITIAL DISTRIBUTION LIST

	No. Copies
1. Defense Documentation Center Cameron Station Alexandria, Virginia 22314	20
2. Library Naval Postgraduate School, Monterey, California	2
3. Ordnance System Command HQTS Department of the Navy Washington, D. C. 20360	1
4. Prof Harold A. Titus Department of Electrical Engineering Naval Postgraduate School, Monterey, California	1
5. LT Ralph E. Hudson FAIRECONRON ONE FPO San Francisco, California 96667	1

DOCUMENT CONTROL DATA - R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) U. S. Naval Postgraduate School Monterey, California	2a. REPORT SECURITY CLASSIFICATION Unclassified
	2b. GROUP

3. REPORT TITLE

FILTERING NONLINEAR MEASUREMENTS

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

Ph D Dissertation, December 1966

5. AUTHOR(S) (Last name, first name, initial)

HUDSON, Ralph E., LT USN

6. REPORT DATE December 1966	7a. TOTAL NO. OF PAGES 130	7b. NO. OF REFS 13
-------------------------------------	-----------------------------------	---------------------------

8a. CONTRACT OR GRANT NO. b. PROJECT NO. c. d.	9a. ORIGINATOR'S REPORT NUMBER(S)
	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)

10. AVAILABILITY/LIMITATION NOTICES

Qualified requesters may obtain copies of this report from DDC.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
-------------------------	----------------------------------

13. ABSTRACT

An algorithm is developed for estimating the state of a linear dynamic system excited by a random sequence. The input data are noisy observations which are nonlinear functions of the state. The estimates are best in the sense of least squared residuals. A significant problem in radar tracking is investigated and the effectiveness of the algorithm verified.

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Filter Nonlinear Observation Radar Tracking Least Squares						

INSTRUCTIONS

1. **ORIGINATING ACTIVITY:** Enter the name and address of the contractor, subcontractor, grantee, Department of Defense activity or other organization (*corporate author*) issuing the report.

2a. **REPORT SECURITY CLASSIFICATION:** Enter the overall security classification of the report. Indicate whether "Restricted Data" is included. Marking is to be in accordance with appropriate security regulations.

2b. **GROUP:** Automatic downgrading is specified in DoD Directive 5200.10 and Armed Forces Industrial Manual. Enter the group number. Also, when applicable, show that optional markings have been used for Group 3 and Group 4 as authorized.

3. **REPORT TITLE:** Enter the complete report title in all capital letters. Titles in all cases should be unclassified. If a meaningful title cannot be selected without classification, show title classification in all capitals in parenthesis immediately following the title.

4. **DESCRIPTIVE NOTES:** If appropriate, enter the type of report, e.g., interim, progress, summary, annual, or final. Give the inclusive dates when a specific reporting period is covered.

5. **AUTHOR(S):** Enter the name(s) of author(s) as shown on or in the report. Enter last name, first name, middle initial. If military, show rank and branch of service. The name of the principal author is an absolute minimum requirement.

6. **REPORT DATE:** Enter the date of the report as day, month, year, or month, year. If more than one date appears on the report, use date of publication.

7a. **TOTAL NUMBER OF PAGES:** The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.

7b. **NUMBER OF REFERENCES:** Enter the total number of references cited in the report.

8a. **CONTRACT OR GRANT NUMBER:** If appropriate, enter the applicable number of the contract or grant under which the report was written.

8b, 8c, & 8d. **PROJECT NUMBER:** Enter the appropriate military department identification, such as project number, subproject number, system numbers, task number, etc.

9a. **ORIGINATOR'S REPORT NUMBER(S):** Enter the official report number by which the document will be identified and controlled by the originating activity. This number must be unique to this report.

9b. **OTHER REPORT NUMBER(S):** If the report has been assigned any other report numbers (*either by the originator or by the sponsor*), also enter this number(s).

10. **AVAILABILITY/LIMITATION NOTICES:** Enter any limitations on further dissemination of the report, other than those

imposed by security classification, using standard statements such as:

- (1) "Qualified requesters may obtain copies of this report from DDC."
- (2) "Foreign announcement and dissemination of this report by DDC is not authorized."
- (3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through _____."
- (4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through _____."
- (5) "All distribution of this report is controlled. Qualified DDC users shall request through _____."

If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indicate this fact and enter the price, if known.

11. **SUPPLEMENTARY NOTES:** Use for additional explanatory notes.

12. **SPONSORING MILITARY ACTIVITY:** Enter the name of the departmental project office or laboratory sponsoring (*paying for*) the research and development. Include address.

13. **ABSTRACT:** Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical report. If additional space is required, a continuation sheet shall be attached.

It is highly desirable that the abstract of classified reports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the information in the paragraph, represented as (TS), (S), (C), or (U).

There is no limitation on the length of the abstract. However, the suggested length is from 150 to 225 words.

14. **KEY WORDS:** Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context. The assignment of links, roles, and weights is optional.

thesH8504

Filtering nonlinear measurements.



3 2768 001 03549 6

DUDLEY KNOX LIBRARY